

AD 740605

RADC-TR-72-2, Volume II
Final Technical Report
January 1972



RASTER PLOTTER SOFTWARE
Volume II - Software Documentation

United Aircraft Corporation

Approved for public release;
distribution unlimited.

**Best
Available
Copy**

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D		
<i>(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)</i>		
1. ORIGINATING ACTIVITY (Corporate author)		2a. REPORT SECURITY CLASSIFICATION
United Aircraft Corporation Hamilton Standard Division Windsor Locks, Connecticut 06096		UNCLASSIFIED
		2b. GROUP
		N/A
3. REPORT TITLE		
RASTER PLOTTER SOFTWARE Volume II - Software Documentation		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates)		
Final Report		
5. AUTHOR(S) (First name, middle initial, last name)		
Rodney T. Boorse David L. Sharp Gerald L. Sperber		
6. REPORT DATE	7a. TOTAL NO OF PAGES	7b. NO OF REFS
January 1972	338	0
8a. CONTRACT OR GRANT NO	9a. ORIGINATOR'S REPORT NUMBER(S)	
F30602-71-C-0107	HSER 5924	
Job Order No. 60350000	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
	RADC-TR-72-2 Volume II (of two)	
10. DISTRIBUTION STATEMENT		
Approved for public release; distribution unlimited.		
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY
None		Rome Air Development Center (IRAG) Griffiss Air Force Base, New York 13440
13. ABSTRACT		
<p>A color coding scheme for hierarchically representing the cartographic symbology associated with JOG series 1501-A charts has been developed and implemented at the RADC Experimental Cartographic Facility (ECF). The scheme employs a three level hierarchy, with up to ten colors available for each level of the hierarchy. The first level of color coding is accomplished on a registered, transparent overlay to the base manuscript. The annotative procedure is straightforward and permits rapid color tagging with very low error raster.</p> <p>Based on this color coding scheme, computer programs which accomplish the necessary recognition and interpretation of these color codes were developed and demonstrated. Color tagged raster data recorded by the RADC Automatic Color Scanning Device comprised the input to these programs, while the output comprised lineal data fully annotated in accordance with the prescribed code structure. In this process, separate scannings are required for the base manuscript and the coded overlay.</p> <p>In addition, computer programs which symbolize pre-formatted raster data for output on the RADC Graphic Plotter were developed and demonstrated. These programs accept edited and symbolized raster line center data, which has been sorted by the assigned final printed graphic color, and generate the aperture and density level commands required to drive the Graphic Plotter.</p>		

DD FORM 1473

1 NOV 65

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14	KEY WORDS	LINK A		LINK B		LINK C	
		ROLE	WT	ROLE	WT	ROLE	WT
	Automated Cartography Computer Programming Faster Data Processing Color Coding						

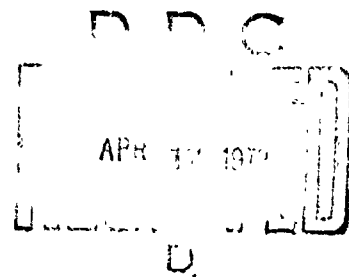
UNCLASSIFIED

Security Classification

RASTER PLOTTER SOFTWARE
Volume II - Software Documentation

Rodney T. Boorse
David L. Sharp
Gerald L. Sperber

United Aircraft Corporation



Approved for public release;
distribution unlimited.

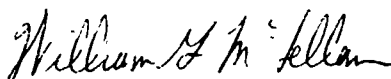
FOREWORD

This report describes the work performed by the Hamilton Standard Division of United Aircraft Corporation, Windsor Locks, Connecticut, in accordance with the terms and requirements of contract F30602-71-C-0107, Job Order Number 60350000, for Rome Air Development Center, Griffiss Air Force Base, New York. William G. McLellan (IRAG) was the RADC Project Engineer administering the program. Secondary report number is HSER 5924.

In addition to the authors, Hamilton Standard personnel contributing to program implementation and the preparation of this report were Messrs. J.R. Byrd, R.G. Hubbard, H.R. Lee, and R.I. Skolnick.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS).

This technical report has been reviewed and is approved.



Approved: WILLIAM G. MCLELLAN
Project Engineer



Approved: FRANZ H. DETTMER
Colonel, USAF
Chief, Intel & Recon Division

FOR THE COMMANDER:



FRED I. DIAMOND
Acting Chief, Plans Office

APPENDIX I

FEATURE IDENTIFICATION SOFTWARE SYSTEM OPERATING INSTRUCTIONS

APPENDIX I

TABLE OF CONTENTS

<u>Paragraph</u>		<u>Page</u>
SECTION I		
DISK PROGRAM CATALOGING PROCEDURE		
1.	GENERAL	73
2.	DECTAPE-TO-DISK LOADING INSTRUCTIONS	73
3.	DISK-TO-CORE PDP-9 MONITOR LOADING INSTRUCTIONS . .	75
SECTION II		
FEATURE IDENTIFICATION SOFTWARE SYSTEM PROGRAM		
1.	GENERAL	77
2.	FISS PROGRAM LOADING INSTRUCTIONS	77
3.	PROGRAM INITIALIZATION AND CONTROL INSTRUCTIONS . .	78
4.	ERROR CONDITIONS AND PROGRAM STOPS	79
5.	PROCESSING STATISTICAL DATA OUTPUTS	80

APPENDIX I
INDEX OF DIAGRAMS

<u>Figure</u>		<u>Page</u>
I-1	FISS Program Run	81

c. Select DECTape-to-Disk Program Loading Options

Place the following PDP-9 DISK CONTROL UNIT panel switches in the "OFF" (down) or "ON" (up) positions at the completion of the DSKSAV loading operation. The operator will be signalled at the completion of this operation by the following system generated administrative message.

'DSKSAV'

Disk control panel settings:

- Control switches 0-29 - "OFF"
- WRITE PROTECT switch - "OFF"

d. Transfer PDP-9 Operating System Object Programs from DECTape-to-Memory-to-Disk File Storage.

(1) System maintenance program, DSKSAV loads object coded executive system programs from DECTape to active core memory. The resulting core image program is then transferred onto disk. To initiate the execution of this function, perform the following sequence of console operations:

- Set console ACCUMULATOR switches to 000000₈
- Depress CONTINUE

(2) Upon the successful completion of system program transfer from DECTape to disk storage, the following administrative message is typed on-line via the console keyboard/page printer:

"DSKSAV"

e. Reset Disk Control Panel Options to Initiate the Loading and Building of an FISS Disk Resident Program File

(1) Place the following PDP-9 Disk Control Unit panel switches in the "OFF" (down) or "ON" (up) positions:

- Control switches 0-29 - "ON"
- WRITE PROTECT switch - "ON"

i. Transfer FISS Object Program from DECtape-to-Memory-to-Disk File Storage

(1) Initiate transfer of FISS Object Program from DECtape 2 to disk file storage as follows:

- Set console ACCUMULATOR switches to 000002₈
- Depress CONTINUE

(2) The following message will be typed on-line when the program load and transfer operation previously described has been completed:

"DSKSAV"

3. DISK-TO-CORE PDP-9 MONITOR LOADING INSTRUCTIONS

a. Load the Disk File Resident MONITOR System Into Active Core Memory

- (1) Place system utility program, DISK, in console paper tape reader
- (2) Set console ADDRESS switches to 77637₈
- (3) Depress the following console function keys in the order specified

below:

- STOP
- I/O RESET
- READIN

(4) Upon the successful completion of the above program loading function, the following system generated administrative message is typed on-line:

"MONITOR V4E"

(5) The above message signals the termination of the operating and applications program system disk loading sequence, and indicates that the core resident MONITOR program is available for use.

SECTION II

FEATURE IDENTIFICATION SOFTWARE SYSTEM PROGRAM

1. GENERAL

The following paragraphs describe the sequence of instructions to load the FISS program into core memory from the PDP-9 disk program catalog, and to initialize the program once control is relinquished by the resident MONITOR. Both the program loading and initialization processes are carried out on-line by way of the PDP-9 console teleprinter and keyboard device. To accomplish the loading process the user interacts with the resident PDP-9 MONITOR system as detailed in Section II.2 of this appendix. Program initialization is accomplished by interacting with the connection control program, CNTRL, and is described in Section II.3.

The following I/O message notation convention is employed in this appendix: MONITOR system and FISS program generated output messages are underscored, as shown in the example below. Keyed input messages, entered by the user, are not underscored.

Example:

System Output Message: MONITOR V4E

Operator Input Message: LOAD

2. FISS PROGRAM LOADING INSTRUCTIONS

a. MONITOR V4E

System generated message to signal the user that the MONITOR system is core memory resident and is available for use.

b. G 2

Enter the above input message to initiate the loading of the Feature Identification Software System Program from Disk Area 2 to primary memory.

c. ↑ S

Enter this message to start execution of program.

Preceding page blank

3. PROGRAM INITIALIZATION AND CONTROL INSTRUCTIONS

This section provides a detailed description of the procedural steps needed to initialize the Feature Identification Software System. The program initialization process is implemented by means of a query-response control mechanism. The on-line teletype serves as the primary communications link between man and machine, and is employed to generate administrative messages instructing the user of the specific data and control parameters needed for normal execution; to generate diagnostic messages signaling the type of keying or procedural errors committed by the user during the initialization process; and to serve as the input medium for user response messages.

All user input messages must be terminated by a carriage return, unless otherwise indicated.

a. MOUNT ACSD MANUSCRIPT TAPE ON DRIVE 3. ↑ P

b. ↑ P

Enter CONTROL P upon completion of above task.

c. MOUNT ACSD OVERLAY TAPE ON DRIVE 5. ↑ P

d. ↑ P

Enter CONTROL P upon completion of above task.

e. MOUNT SCRATCH TAPE ON DRIVE 6. ↑ P

f. ↑ P

Enter CONTROL P upon completion of above task.

g. ENTER RESOLUTION AS 2, 4, 5 or 6.

h. N ↑ P

Enter ACSD resolution increment value N (decimal integer = 2, 4, 5 or 6) at which input data was generated.

NOTE: Upward or downward scaling of the output data file will occur if a resolution increment other than the original ACSD machine setting is entered.

i. ENTER MIN AREA SETTING AS XX ELEMENTS.

j. Enter any odd 2-digit number between 01 and 31 to indicate original ACSD MIN AREA machine setting, e. g. 03 ↑ P.

k. ENTER LINEAL NEIGHBORHOOD RANGE AS XX UNITS.

l. Enter a two-digit number greater than zero (e. g. 05 ↑ P) to indicate the maximum number of resolution elements to be used to relate lineal data points along the scan axis. Parameter is not requested if a MIN AREA of 1 was indicated.

m. ENTER AREAL NEIGHBORHOOD RANGE AS XX UNITS.

n. Enter a two-digit number greater than zero (e. g. 02 ↑ P) to indicate the maximum number of resolution elements to be used to relate areal data points along the scan axis.

o. ENTER MINIMAL FEATURE LENGTH AS XX RASTER UNITS.

p. Enter a two-digit number (XX) between 01 and 31 to indicate the minimum number of raster data points in a feature data set for that data set to be retained, e. g. 15 ↑ P.

q. ENTER GAP PARAMETER AS X RASTER UNITS.

r. Enter a one-digit octal number greater than or equal to zero (e. g. 1 ↑ P) to indicate the maximum number of raster scan lines to span to establish neighborhood connection.

s. ENTER BASE COLOR CODE AS XX.

t. Enter a two-digit decimal number between 00 and 15 to indicate the color code of the desirable base manuscript feature. The color codes are the ACSD machine settings used for that specific base manuscript and its associated overlay.

4. ERROR CONDITIONS AND PROGRAM STOPS

If an abnormal condition occurs during the execution of the program, one of the following messages will be printed on-line via the console teleprinter.

a. STOP 000001

ACSD Tape Error - Magnetic tape error encountered during ACSD base manuscript tape generation.

b. STOP 000002

PDP9GE Overflow - Data point encountered that exceeds the limits of CDP system. Rerun program. If error persists, dump disk and examine data set point values for possible error.

c. STOP 000003

ACSD Tape Error - Magnetic tape error encountered during ACSD overlay tape generation.

d. STOP 000004

Line Buffer Overflow-ACSD scan line record encountered on base manuscript tape that is larger than the fixed sized raster record input buffer.

e. STOP 000005

Core Memory Overflow - More than 128 feature data sets simultaneously active in memory. Reload program and increase the range parameters in input messages (k) and (m).

f. STOP 000006

Disk Overflow - Available disk storage exhausted. Reload connection program increase or decrease neighborhood range parameters (see input messages k and m).

g. STOP 000007

Line Buffer Overflow - ACSD scan line record encountered on overlay tape that is larger than the fixed sized raster record input buffer.

5. PROCESSING STATISTICAL DATA OUTPUTS

Immediately following the successful completion of the line connection process, timing and data statistics are printer on-line via the console teleprinter. These include the number of points input and output, convergences and divergences of features, linear/arcual transitions encountered, feature data sets output, and so on. Figure I-1 contains a typical example of the statistical data generated.

MONITOR V4E

\$G 2

1S

MOUNT ACSD MANUSCRIPT TAPE ON DRIVE 3.1P

1P

MOUNT ACSD OVERLAY TAPE ON DRIVE 5.1P

1P

MOUNT SCRATCH TAPE ON DRIVE 6.1P

1P

ENTER RESOLUTION AS 2,4,5,OR 6. >

41P

ENTER MIN AREA SETTING AS XX ELEMENTS. >

011P

ENTER AREAL NEIGHBORHOOD RANGE AS XX UNITS. >

011P

ENTER MINIMAL FEATURE LENGTH AS XX RASTER UNITS. >

101P

ENTER GAP PARAMETER AS X RASTER UNITS. >

11P

ENTER BASE COLOR CODE AS XX. >

091P

RASTER-TO-LOCUS CONVERSION STATISTICS

NUMBER OF POINTS INPUT =	2835.	
NUMBER OF POINTS OUTPUT =	2787.	
NUMBER OF CONVERGES =	30,	
NUMBER OF DIVERGES =	31	
NUMBER OF LINEAR/AREAL TRANSITIONS =		17
NUMBER OF FEATURE DATA SETS =	137	
NUMBER OF FEATURES UNDER MINIMAL LENGTH =		51
NUMBER OF DISK SEGMENTS USED =	288	
APPROXIMATE LENGTH OF PRODUCT =	11.148	
RUN TIME =	1 MIN. 12 SEC.	

MONITOR V4E

\$

FIGURE I-1 . FISS PROGRAM RUN

APPENDIX II

FEATURE IDENTIFICATION SOFTWARE SYSTEM PROGRAM DESCRIPTION

APPENDIX II

TABLE OF CONTENTS

<u>Paragraph</u>		<u>Page</u>
	SECTION I	
	FEATURE IDENTIFICATION SOFTWARE SYSTEM PROGRAM	
1	GENERAL	83
2	SYSTEM DESCRIPTION ,	83
	a. Object Program Deck Arrangement	83
	b. System Diagrams	82
	c. Logic Diagram Symbology	84
	SECTION II	
	DESCRIPTION OF SUBPROGRAMS	
1	NEW FISS PROGRAMS	89
	a. UPDATE (Update Mother Data Set)	89
	b. DCDCOL (Decode Color)	90
	c. COLCLR (Collect Color)	90
	d. FILBUF (Fill Line Buffer)	91
2	MODIFIED CAST PROGRAMS	93
	a. CNTRL (Feature Identification Software System Control Program)	93
	b. TRMDS (Terminate Data Set)	97
	c. EXTRCT (Data Element Extraction)	98
	d. ALINE (Intersection Resolution)	99

SECTION II (CONT)

<u>Paragraph</u>	<u>Page</u>
3	OBJECT-TIME SYSTEM AND SCIENCE LIBRARY ROUTINES 101
a.	Object-Time System Routines 101
b.	Science Library Routines 102

SECTION III FISS FLOW DIAGRAMS

INDEX OF DIAGRAMS

<u>Figure</u>	<u>Page</u>
II-1	System Logic Flow Diagram 85
II-2	Feature Identification Software System Block Diagram 86
II-3	Logic Diagram Symbology 87
II-4	Flow Diagram of UPDATE 104
II-5	Flow Diagram of DCDCOL 105
II-6	Flow Diagram of COLCLR 106
II-7	Flow Diagram of FILBUF 108
II-8	Flow Diagram of CNTRL 110
II-9	Flow Diagram of TRMDS 114
II-10	Flow Diagram of EXTRCT 117
II-11	Flow Diagram of ALINE 118

SECTION I

1. GENERAL

The system software elements described in this appendix consists of the new subprograms required and the CAST Engineering Change "A" programs modified to implement the Feature Identification Software System (FISS). The four new programs are detailed in Section II. Of the programs resulting from the CAST Engineering Change "A" effort, FILBUF required extensive changes and is detailed in Section II.1, while four other programs required only minor modifications which are described in Section II.2. All other CAST programs remain as originally documented. Section II.3 identifies the PDP-9 Object-Time System programs and Science Library macro sub-routines required for various Fortran IV I/O and mathematical functions. Section III contains the flow diagrams of the FISS programs described in Section II.1 and II.2.

2. SYSTEM DESCRIPTION

The FISS Program is a self-contained, stand-alone program designed to be executed on a PDP-9 processing system having the following configuration:

<u>Quantity</u>	<u>Equipment Item Description</u>
1	DEC PDP-9 Control Processing Unit (CPU)
8	4096 18-bit Word Core Memory Modules
1	10 CPS, Console Keyboard/Teleprinter
3	7-Track, 45IPS, 556/800 BPI IBM Compatible Magnetic Tape Units W/Controller
4	DECtape Units W/Controller
1	DEC PDP-9 Paper Tape Reader
1	DEC PDP-9 Paper Tape Punch
1	1 Million 18-Bit Word Disk File W/Controller
1	ANELEX, 1000 LPM Line Printer

a. Object Program Deck Arrangement

The FISS Program is assembled and compiled as a "relocatable" program. With the exception of the control program (CNTRL), which by definition must be the first program in the object deck, the remaining object programs may be arranged in any order.

b. System Diagrams

Figure II-1 shows the control logic flow for the FISS program. PDP-9 OTS

and Science Library subprograms and routines are not shown since they are loaded and enter the procedural flow only at the time of execution. As shown, the program as designed is divided into three functional phases of execution: Initialization and Phase I and II.

(1) Initialization

Package consists of the worker programs shown in addition to PDP-9 OTS input/output handlers and drivers loaded at time of execution. These provide a software controlled user-system interface as is required either to enter needed control information and data, or to output system generated, administrative messages via the console keyboard/teleprinter devices.

(2) Phase I

Software is devoted to the enactment of the point-to-point correlation functions as performed by the software modules indicated. Such functions are performed in a serial batch mode of operation. Overlay color information is correlated to base manuscript positional data to establish symbolic color-coded feature header information. Lineal formatted output data records are retained in disk file storage until a base color data connection pass is completed. Such data is then made available to the Phase II processor for final output formatting.

(3) Phase II

Consists of those translatory and record data formatting functions as required to transform internally formatted data records to the required standard ACS Manuscript (MMS) Tape File Format. This processor, like the previous Phase I processor, is designed to enact these functions in a serial batch processing mode.

Figure II-2 shows the overall functional block diagram of the FISS program.

c. Logic Diagram Symbolology

The syllabus of logic diagram symbolology, as shown in Figure II-3, defines the program flow diagram notation conventions employed in this appendix.

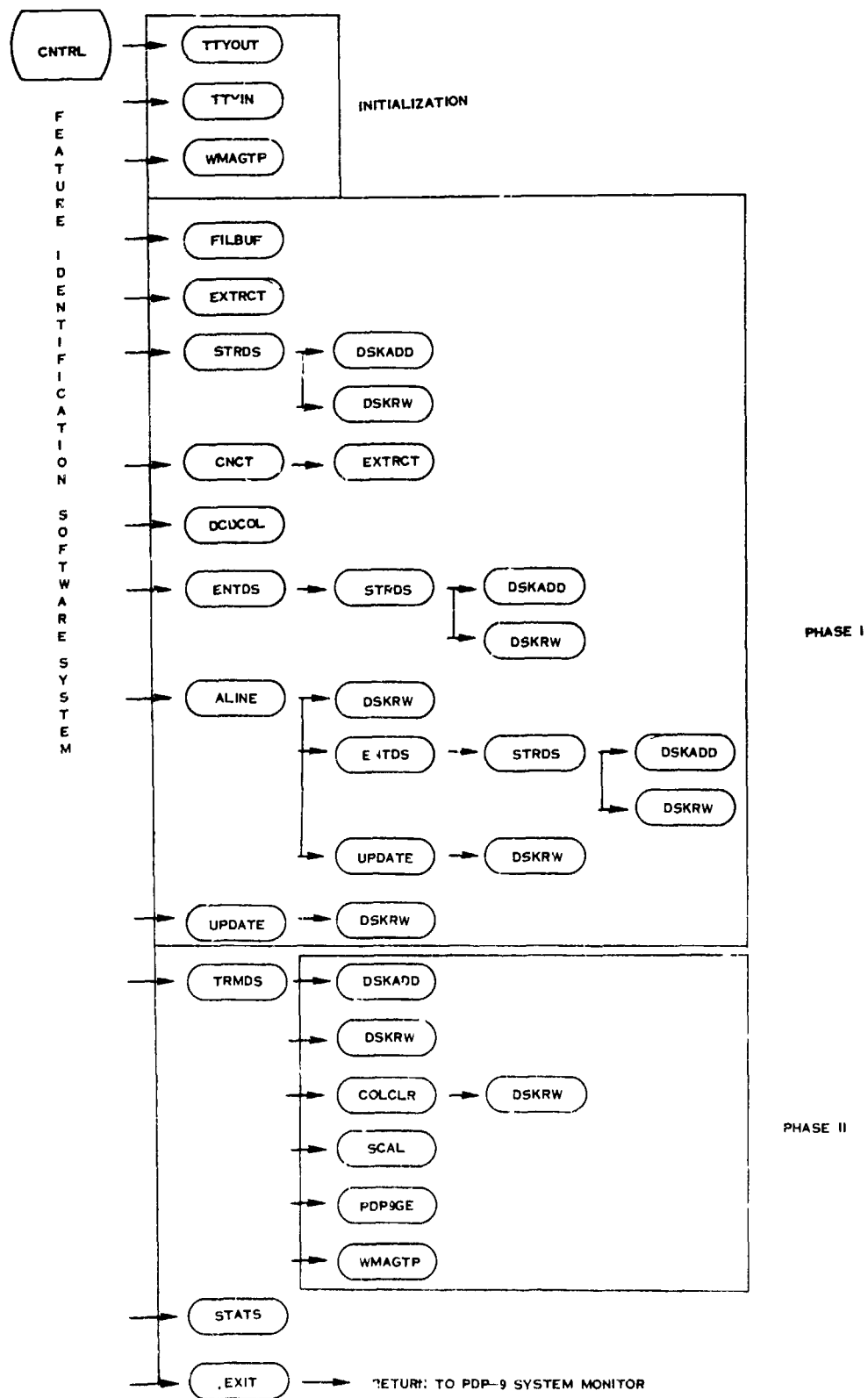


FIGURE II-1. SYSTEM LOGIC FLOW DIAGRAM

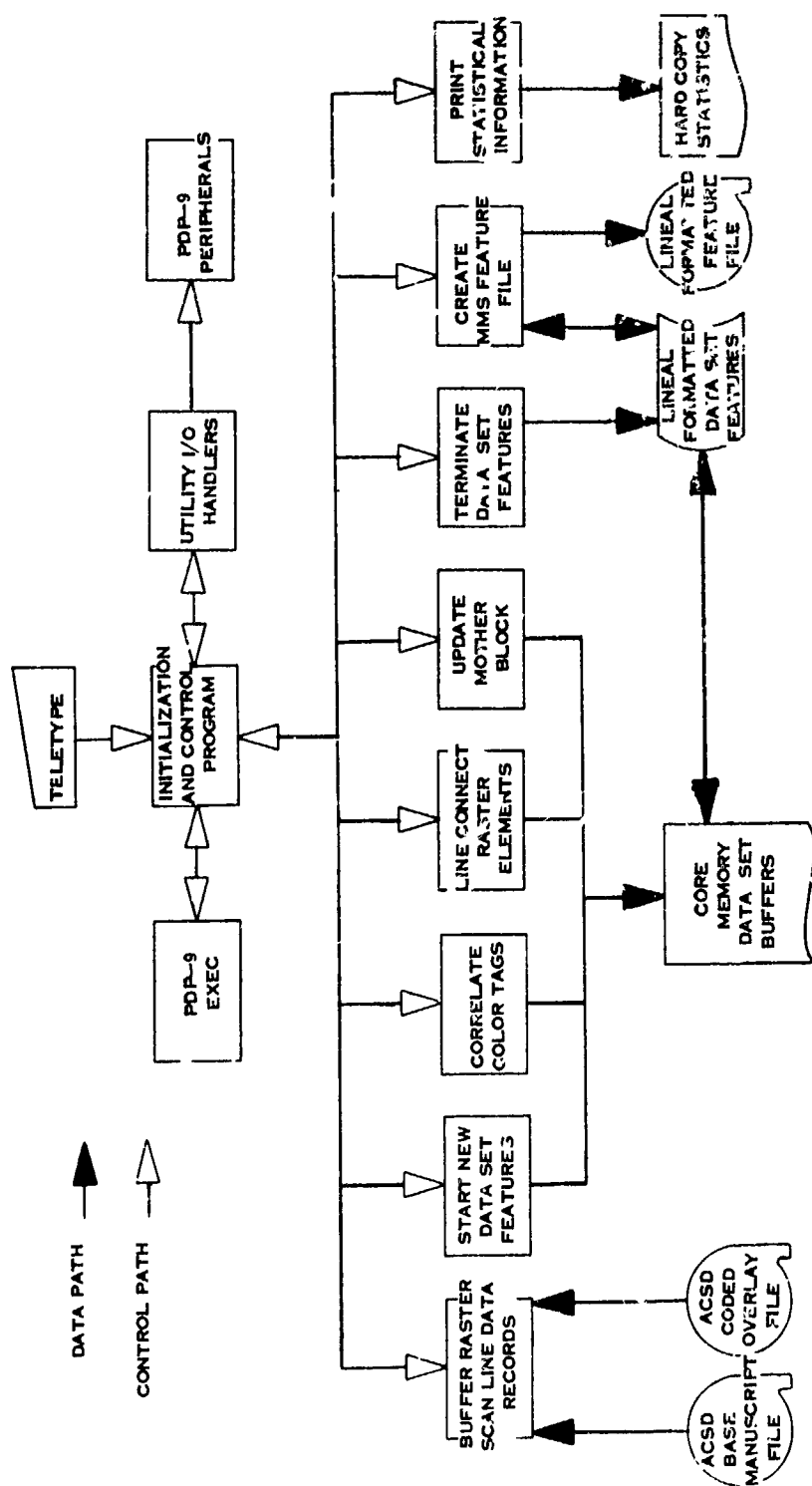


FIGURE II - 2. FEATURE IDENTIFICATION SOFTWARE SYSTEM BLOCK DIAGRAM

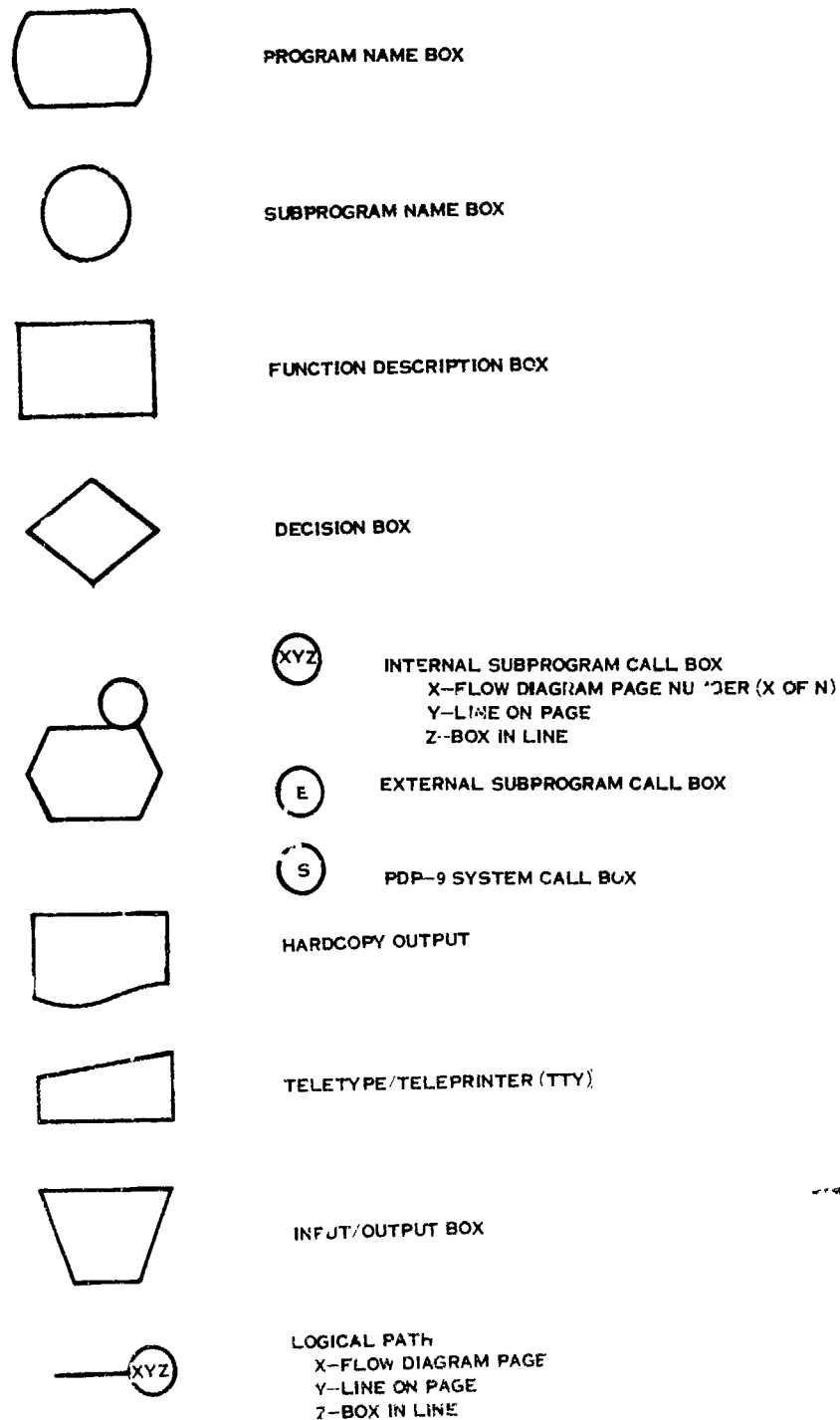


FIGURE U-3. LOGIC DIAGRAM SYMBOLOGY

SECTION II

DESCRIPTION OF SUBPROGRAMS

1. NEW FISS PROGRAMS

a. UPDATE (Update Mother Data Set)

(1) Abstract

UPDATE updates the mother data set block identifying a feature with data set linkage and color identification information.

(2) Program Description

Upon unconditional data set termination UPDATE retrieves the specified mother data set block from disk, if not already core-resident, updates correlation pointer no. 2 with data set linkage information, overwrites a data set header no longer used with the core-resident color identification word describing the data set, and rewrites the mother data set block to disk. In the case of core resident mother data set blocks, the initial disk read is bypassed. The update information is used by COLCLR for feature header formulation.

(3) Inputs

Calling Sequence -

L	LAC	DSCAD
L + 1	JMS*	UPDATE
L + 2	- -	Next sequential instruction

where:

DSCAD = core address of data set block.

(4) Outputs

Disk -

Updated mother data set block.

(5) Referenced Subprograms

DSKRW

Preceding page blank

b. DCDCOL (Decode Color)

(1) Abstract

DCDCOL updates the specified core-resident data set block color word with color identification information.

(2) Program Description

DCDCOL uses the contents of the ACC (color code) to compute a table address of the corresponding bit map representation for that color code. The data set requiring a color identification update is identified via the data set address passed in the calling sequence. For each call to DCDCOL the current color word for the specified data set is logically ORed with the bit map representation of the color code identified in the calling sequence.

(3) Inputs

Calling Sequence -

L		LAC	COLOR
L + 1		JMS*	DCDCOL
L + 2		SKP	Return location
L + 3	ARG	Ø	
L + 4			Next sequential instruction

where;

ARG = core address of data set block.

(4) Outputs

Memory -

Updated data set color word.

(5) Referenced Subprograms

None

c. COLCLR (Collect Color)

(1) Abstract

COLCLR is the Raster Plotter Color Code Implementation routine that collects all the color codes from all the data sets associated with a feature in

order to formulate the proper header for that feature.

(2) Program Description

COLCLR is called by TRMDS whenever the correlation between the data sets of a feature and the overlay color codes has been completed. COLCLR chains through all the data sets associated with a given feature, extracting the required color code from each data set as it goes along. The data sets are read from disk using DSKRW. Required information from the data block headers is extracted by the internal routine HEADER. Another internal routine, GETHDR, takes the extracted color code information, separates out the proper color code and formulates the appropriate word for the feature header identification.

(3) Inputs

Calling Sequence -

L		JMS*	COLCLR
L + 1		JMP	.+3
L + 2		.DSA	ARG1
L + 3	ARG2	Ø	
L + 4		Next sequential instruction

where:

ARG1 = mother block of an end of the feature for which the color information is to be collected.

ARG2 = the header word returned to the calling program.

(4) Outputs

Memory -

Color code word that uniquely identifies the type of feature interested in.

(5) Referenced Subprograms

DSKRW

d. FILBUF (Fill Line Buffer)

(1) Abstract

FILBUF buffers raw ACSD records from magnetic tape to an internal buffer and transfers logical scan line records and its associated color information from the tape buffer to a specified scan line memory buffer and its associated color identifi-

ication buffer. The raster scan value (y - coordinate) is returned to the calling program.

(2) Program Description

FILBUF is called whenever a line buffer is to be filled. Each call fills the specified line buffer with coordinate data of the next consecutive scan line. The coordinate data is followed by a trailer word indicating the end of coordinate data. Associated with each data element in the line buffer is an ACSD color code in the associated color identification buffer identifying the color code representing that particular data element.

FILBUF recognizes three types of raster elements appearing in the raw ACSD records. Line center and areal data are directly transferred from the tape to the specified line buffer. Min Area data is conditionally transferred to the specified line buffer. For a Min Areas of one, Min Area data is interpreted as noise and is not passed to the line buffer as valid data. For Min Area settings other than one, the Min Area data points are converted to a begin area point, and an end area point is generated using the begin area plus the Min Area setting minus one. If the line buffer capacity (1024 words) is reached, line buffer overflow has occurred, otherwise the y-coordinate associated with the x-coordinate data is returned to the calling program. An end-of-data condition is indicated to the calling program by a return to the location following the call arguments address.

If line buffer overflow occurs, a terminal condition is reached, and the program is aborted causing the following message to be output on the teleprinter: "STOP 000004"(STOP 000007). If a tape error was encountered during ACSD tape generation, an error condition is recognized and the program aborted causing the following message to be output on the teleprinter: "STOP 000001"(STOP 000003).

(3) Inputs

(a) Calling Sequence -

L	JMS*	FILBUF	
L + 1	JMP	.+3	Return location
L + 2		ARG	
L + 3	JMP	EOD	End-of-data return location
L + 4	--		Next sequential instruction

where:

ARG = address of buffer arguments

where buffer arguments:

ARG1 = address of line buffer address

ARG2 = y-coordinate of line in buffer

(b) Magnetic Tape -

Raw ACSD multi-color file data records (1024 24-bit words).

(4) Outputs

(a) Memory -

Scan line buffers of raster data elements and associated color identification buffers.

(b) Teleprinter -

The following message is output when line buffer overflow (base manuscript tape) occurs: "STOP 000004".

The following message is output when line buffer overflow (overlay tape) occurs: "STOP 000007".

The following message is output when ACSD tape error (base manuscript tape) is encountered: "STOP 000001".

The following message is output when an ACSD tape error (overlay tape) is encountered: "STOP 000003".

(5) Referenced Subprograms

None

2. MODIFIED CAST PROGRAMS

a. CNTRL (Feature Identification Software System Control Program)

(1) Abstract

CNTRL is the supervisory program element of the Feature Identification Software System. Following system initialization the program buffers raster scan records from raw ACSD tape files by calling FILBUF, locates and extracts data elements and associated color code information from line buffers by calling EXTRCT,

and correlates the data elements to active feature data sets by calling the line connection algorithm. Feature data set initialization and construction are performed by STRDS and ENTLIS respectively. MMS tape generation is initiated by TRMDS upon successful completion of the feature data set generation. Statistical information compiled during program execution is output via the teleprinter and control is returned to the system MONITOR.

(2) Program Description

During system initialization CNTRL receives via the on-line teletype the specific input parameters which are to be used to process the selected ACSD tapes. Feature line segments are represented as data set blocks (mother block and its associated daughter blocks). Each raster scan element (lineal or areal) is mapped into each active data set in search of a neighborhood correlation. If a raster element meets the neighborhood criteria of an active data set, it is entered into that active data set, otherwise the raster element is used to create a new data set. As raster elements in the scan line buffer are exhausted, a new scan line record is introduced. As a data set block becomes full, the block is catalogued and transferred from active memory to disk for retrieval upon final termination of the line connection process.

As feature data sets are created from positional data represented on the base manuscript, symbolic color data represented on the color overlay is correlated to the feature data sets. Color overlay data is compiled for each feature data set. Upon unconditional feature data set termination, the color information is included in the mother data set block of the feature data set.

A data set is not immediately terminated if no raster element can be associated to that data set for a particular scan line. This data set remains active for N scan lines as requested during system initialization by the gap parameter. If N scan lines are processed without a coordinate update of an active data set, the data set is unconditionally terminated.

Conditional data set termination occurs when a convergent data condition is recognized by CNTRL, i. e., when multiple data sets are related to a single raster element. When convergence occurs, the converging data sets are conditionally terminated and the object raster element is used to create a new data set which is marked as a convergent data set and is linked to the above related data sets. The convergent data set is eventually used to define the intersection point of the convergent data. Dependent upon the value of the minimal feature length parameter entered via the teletype during system initialization the convergent data set completely defines the intersection point (minimal) or is of valid length to be a distinct line segment feature of its own. Once the intersection point is established, it is entered as the last point of its associated conditionally terminated data sets which are then unconditionally terminated. If two data sets converge into a single data set which is under minimal feature length, the two data sets remain correlated to each other. If either more than

two data sets converge into a single data set or the convergent data set is of minimal feature length, the convergent condition correlations are erased.

Divergent data conditions (maxima), i. e., when multiple raster elements are related to a single active data set, also require the resolution of an intersection point. In this case the single active data set is the divergent data set and is used to resolve the intersection point of the divergent data. Again, if the data set is under minimal feature length, the data is not a valid feature and is used only to resolve the intersection point which is used to create the new data sets defined by the divergent raster elements. If the divergent data set is under minimal feature length and only two data sets diverge from it, the two data sets are correlated to each other. If either the divergent data set is of minimal feature length or more than two data sets diverge from the divergent data set, the divergent condition correlations are not created.

All data sets which are unconditionally terminated, uncorrelated to any other data sets, and are under minimal feature length are considered to be noise and are not transferred to disk as a valid line segment feature. All other data sets constitute valid line segment features and are transferred to disk.

When line connection is completed CNTRL calls TRMDS to create a lineal file tape in MMS format. Color information compiled during feature data set generation is used to create the feature header record. Statistical information compiled during execution of the line connection function is output via the teleprinter upon successful program completion.

(3) Inputs

(a) Calling Sequence -

None - stand alone program

(b) Keyboard Entries -- Operator Input Response

"↑P"

"X↑P"

"XX↑P"

(c) Magnetic Tape -

1 raw ACSD base manuscript file.

1 raw ACSD color overlay file.

(4) Outputs

(a) Teleprinter -

Program generated query and administrative messages:

"MOUNT ACSD MANUSCRIPT TAPE ON DRIVE 3. ↑P"

"MOUNT ACSD OVERLAY TAPE ON DRIVE 5. ↑P"

"MOUNT SCRATCH TAPE ON DRIVE 6. ↑P"

"ENTER RESOLUTION AS 2, 4, 5, or 6.> "

"ENTER MIN AREA SETTING AS XX ELEMENTS.> "

"ENTER LINEAL NEIGHBORHOOD RANGE AS XX UNITS.> "

"ENTER AREAL NEIGHBORHOOD RANGE AS XX UNITS.> "

"ENTER MINIMAL FEATURE LENGTH AS XX RASTER UNITS.>"

"ENTER GAP PARAMETER AS X RASTER UNITS.> "

"ENTER BASE COLOR CODE AS XX.> "

Program generated statistical information.

(b) Magnetic Tape -

1 lineal formatted feature file tape.

(5) Reference Subprograms

CNCT
FILBUF
STRDS
ENTDS
TRMDS
EXTRCT
ALINE
WMAGTP
STATS
DSKRW
TTYOUT
TTYIN
DCDCOL
UPDATE

b. TRMDS (Terminate Data Set)

(1) Abstract

TRMDS creates the lineal formatted MMS data file. Once line connection is completed, the subprogram is called by CNTRL to transfer all the line segment features residing on disk to magnetic tape in MMS format. Multiple line segment features (mother data set block and 0 to N daughter data set blocks) may constitute an actual feature due to the nature of data set generation at points of maximum and minima. As each data set block is retrieved, TRMDS unpacks the coordinates, scales the coordinates by calling SCAL, converts the coordinates to MMS format and combines any multiply segmented features into a single line feature into a single line feature. TRMDS calls COLCLR to collect all the color information associated with a single line segment feature. Following the header information the coordinate data are transferred to magnetic tape by calling WMAGTP. TRMDS only processes the data existing between the first and last data set block allocated.

(2) Program Description

TRMDS uses DSKADD and DSKRW to search the disk in search of mother blocks. Once a mother block and its associated daughter blocks are located, their headers are checked for any correlation, i. e., another mother-daughter linkage which defines a continuation of the current line segment feature. If a correlation exists the correlated line segment feature becomes the current line segment feature and the above process is repeated until all the line segment features defining a single feature are known and the endpoint of the line feature is reached, or in the case of a closed feature, the correlation returns the chaining of line segment features to the original data set. If no correlation exists the subprogram immediately creates the MMS records describing the line segment feature.

All mother blocks used in creation of a single line feature are marked as used so that they will not again be processed as the disk is searched sequentially. Since all data set features are built with y-coordinates increasing in value, TRMDS processes every other line segment feature defining a single line in an inverse direction, i. e., at points of maxima and minima the function describing the locus path changes in direction. Color information compiled for each feature data set and collected by COLCLR is used to create the feature header record. All coordinate data is unpacked, scaled, converted to GE floating format and written to magnetic tape. TRMDS also accumulates statistical information which is transferred to STATS to be output via the teleprinter.

(3) Inputs

Calling Sequence -

L	JMS*	TRMDS	
L + 1	SKP		Return loca- tion
L + 2		ARG	
L + 3	- -		Next sequen- tial instruction

where:

ARG = address of data set argument.

where data set argument.

ARG1 = core address of data set block.

(4) Outputs

Magnetic Tape -

Lineal formatted feature data sets via subprogram WMAGTP.

(5) Referenced Subprograms

DSKRW
DSKADD
WMAGTP
PDP9GE
SCAL
COLCLR

c. EXTRCT (Data Point Extraction)

(1) Abstract

EXTRCT extracts raster elements (lineal and areal) from a scan line buffer.

(2) Program Description

EXTRCT extracts the next raster element in the specified line buffer and returns the raster element (one word if lineal; two words if areal) to the calling programs. EXTRCT also returns the color code associated with the line buffer entry or entries. Upon exit from the subprogram the value of the accumulator indicates the type of raster element returned. An end-of-line-buffer condition is indicated to

the calling program by a return to the location following the call arguments address.

(3) Inputs

Calling Sequence -

L	JMS*	EXTRCT	
L + 1	JMP	.+3	Return location
L + 2		ARG	
L + 3	JMP	EOD	End-of-data-return
L + 4	- -		Next sequential instruction

where:

AKG = address of buffer arguments.

where buffer arguments:

ARG1 = current line buffer address.

ARG2 = first data point (line center or begin area).

ARG3 = second data point-end area (areal data only).

ARG4 = entry or entries color code.

(4) Outputs

Memory -

Raster elements and element type (via ACC) to the calling program.

(5) Referenced Subprograms

None

d. ALINE (Intersection Resolution)

(1) Abstract

ALINE is called to resolve the intersection point of convergent and divergent data. The minimal feature length is used to determine whether the incoming

data set (divergence) or the outgoing data set (convergence) is of valid length to be a distinct feature of its own or is used only to resolve the intersection.

(2) Program Description

ALINE is called whenever a divergent or convergent data transition occurs. In either case a common intersection point must be defined. In the convergent case resolution of the intersection point is delayed for N scan lines until the outgoing data set is of valid length or is terminated. When either occurs the convergent intersection point becomes the last data entry in all incoming data sets which are then terminated and the first data entry in the outgoing data set, otherwise the outgoing data set is eliminated from the data base. For the divergent case resolution of the intersection point is immediate since the incoming data set contains the required data for resolution. If the incoming data set is of valid length, the divergent intersection point becomes the first data entry in all outgoing data sets and the last data entry in the incoming data set, otherwise the incoming data set is eliminated from the data base. The mother data set block of any data set feature of legitimate length is updated with color and linkage information via UPDATE.

(3) Inputs

Calling Sequence -

L	JMS*	ALINE	
L + 1	SKP		Return location
L + 2		ARG	
L + 3	- -		Next sequential instruction

where:

ARG = address of data set arguments.

where data set arguments:

ARG1 = core address of data set block.

NOTE: Contents of ACC indicate convergent or divergent data.

(4) Outputs

Memory -

Data set blocks with resolved intersection points.

(5) Referenced Subprograms

DSKRW
ENTDS
UPDATE

3. OBJECT-TIME SYSTEM AND SCIENCE LIBRARY ROUTINES

a. Object-Time System Routines

The following is a list of PDP-9 FORTRAN IV Object-Time System macro-subroutines that are loaded at execution time. The compiled FORTRAN programs which contain I/O statements include output calls in the form of globals to various system I/O handler routines. When the compiled object program is loaded, the compiler defined global symbols are utilized to search the FORTRAN library for the needed I/O handler routines. These are loaded into memory and the proper program control linkage established by the Link Loader. Detailed description for these programs are provided in the following DEC PDP-9 manual "Advanced Software System Programmer's Reference Manual"; order number DEC-9A-KFZA-D.

<u>Program Name</u>	<u>Function</u>	<u>Ref. Manual</u>
FIOPS	FIOPS provide the necessary call to IOPS (input-Output Processing System) required by all FORTRAN I/O statements.	P. 11-10
OSTER(.ER)	To announce an error on the teletype.	P. 11-18
TIME	To provide the ability to record elapsed time in minutes and seconds.	P. 11-21
BCDIO	To process formatted READ/WRITE statements in FORTRAN IV programs and sub-programs.	P. 11-13
.SS	To calculate the array element address.	P. 11-12
STOP(.ST)	To process the STOP statement and return control to the monitor.	P. 11-15
SPMSG(.SP)	To print the octal number coded with STOP.	P. 11-17

b. Science Library Routines

The following PDP-9 Science Library mathematical routines are utilized by the FISS programs. These programs are described in the DEC PDP-9 manual referenced in Section II, Subsection 3 .a above.

<u>Routine Name</u>	<u>External Calls</u>	<u>Ref. Manual</u>
ABS	.DA, REAL	P. III-5
FLOAT	.DA, REAL	P. III-5
SIGN	.DA, REAL	P. III-5
.DA(General Get Argument)	None	P. III-8
INTEGE(Integer Arithmetic)	REAL	P. III-8
REAL (Real Arithmetic)	None	P. III-8
.CB (Short Get Argument)	None	P. III-9

SECTION III

FISS FLOW DIAGRAMS

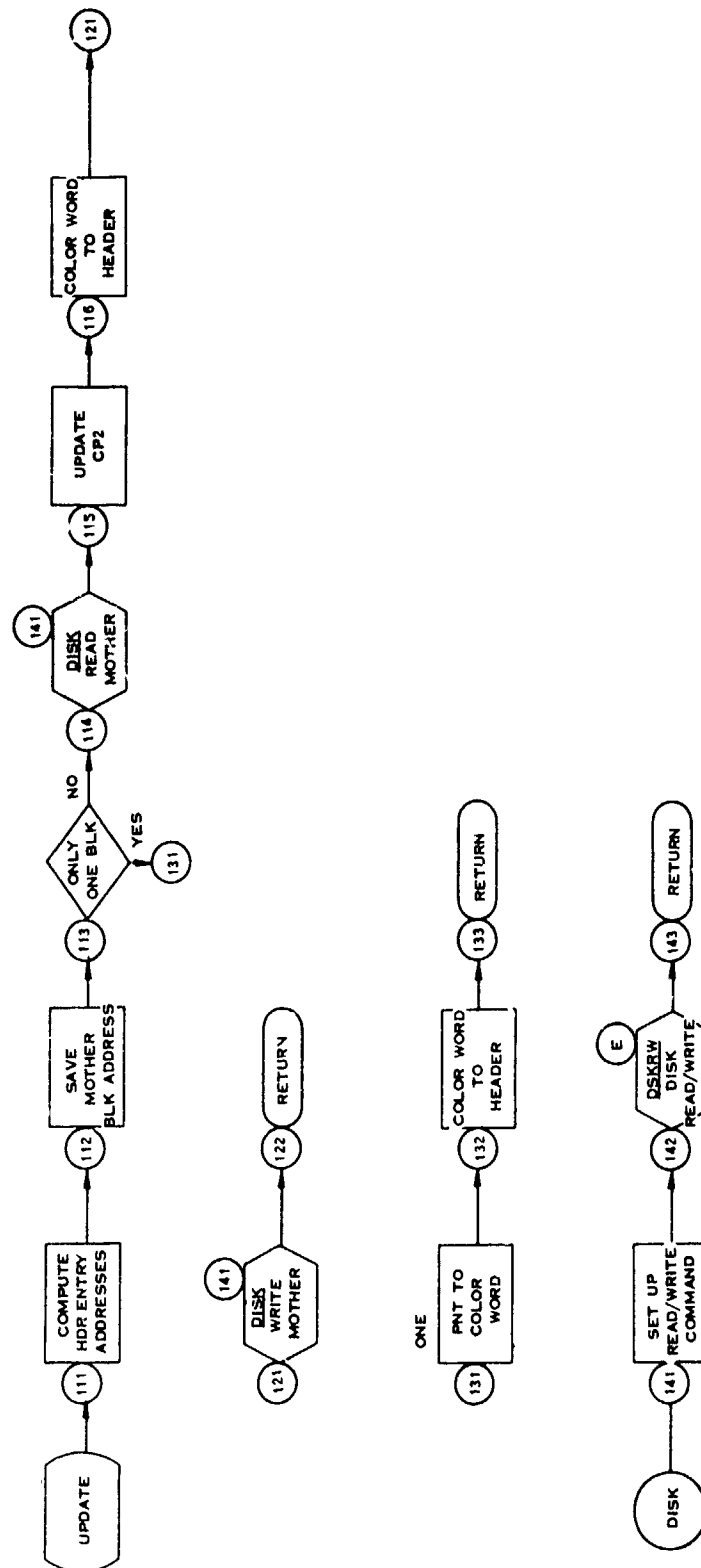


FIGURE II - 4. FLOW DIAGRAM OF UPDATE

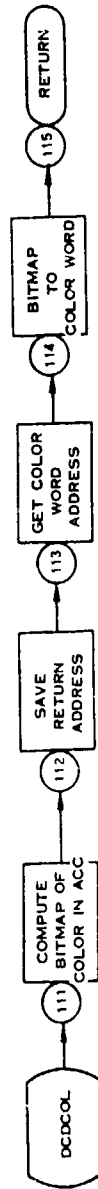


FIGURE 11 - 5. FLOW DIAGRAM OF DCDCOL

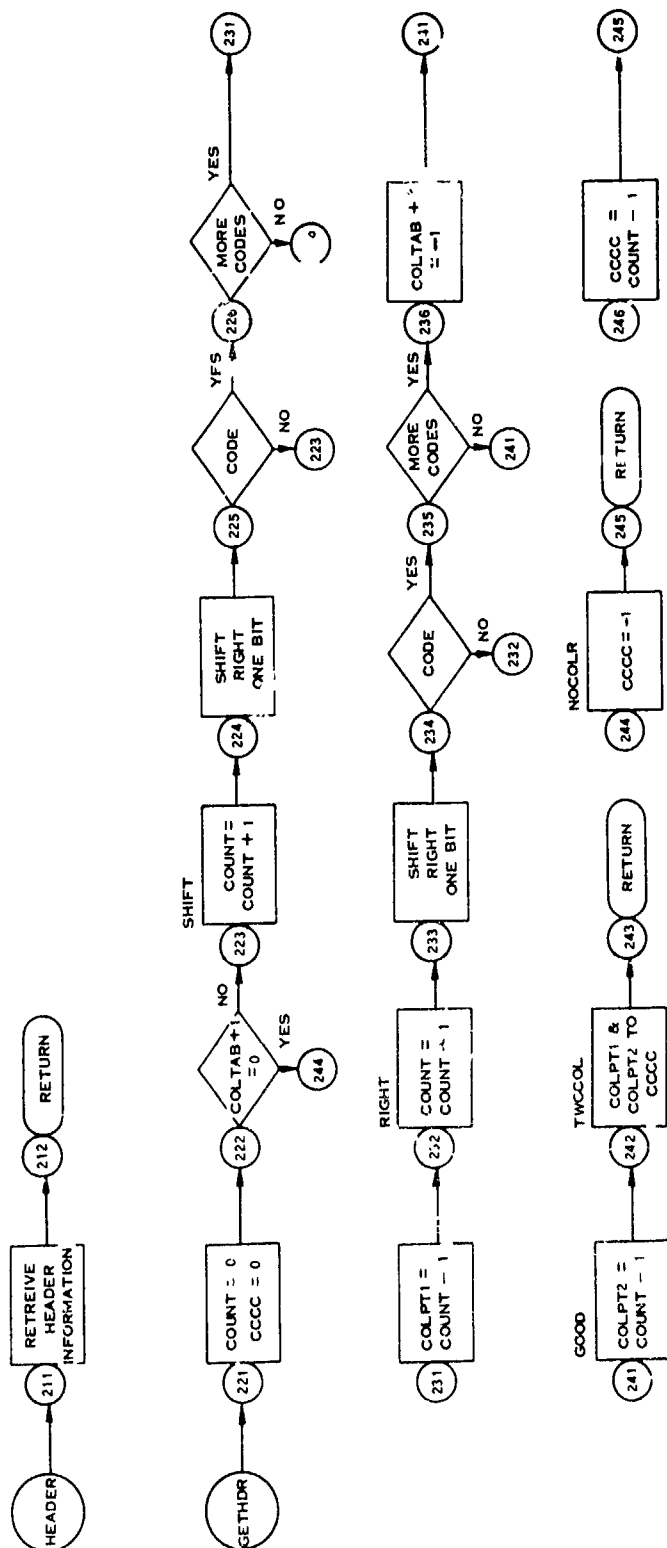


FIGURE II - 6. FLOW DIAGRAM OF COLCLR (CONCLUDED)



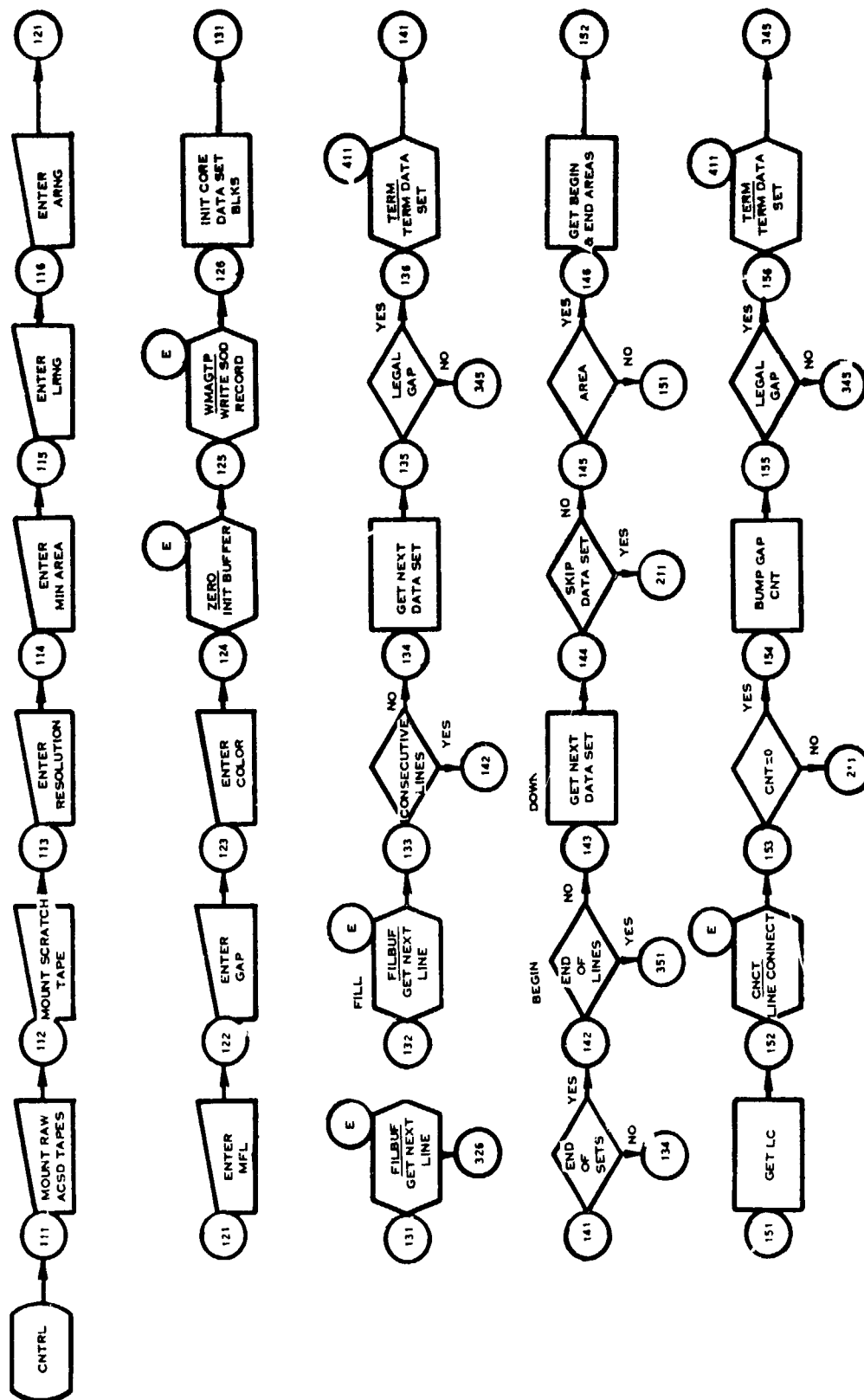


FIGURE II - 8. FLOW DIAGRAM OF CNTRL

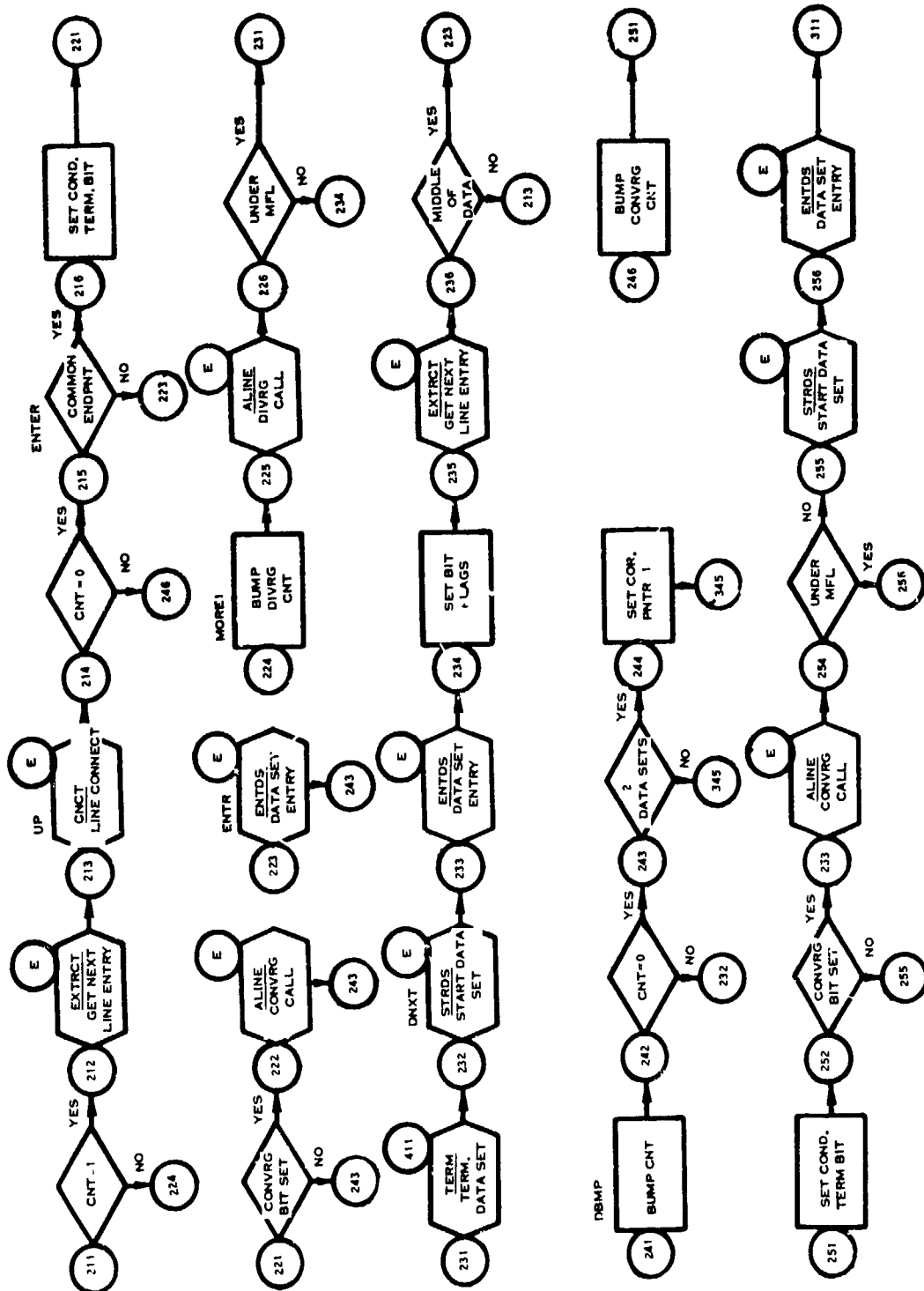


FIGURE II - 8. FLOW DIAGRAM OF CNTRL (CONTINUED)

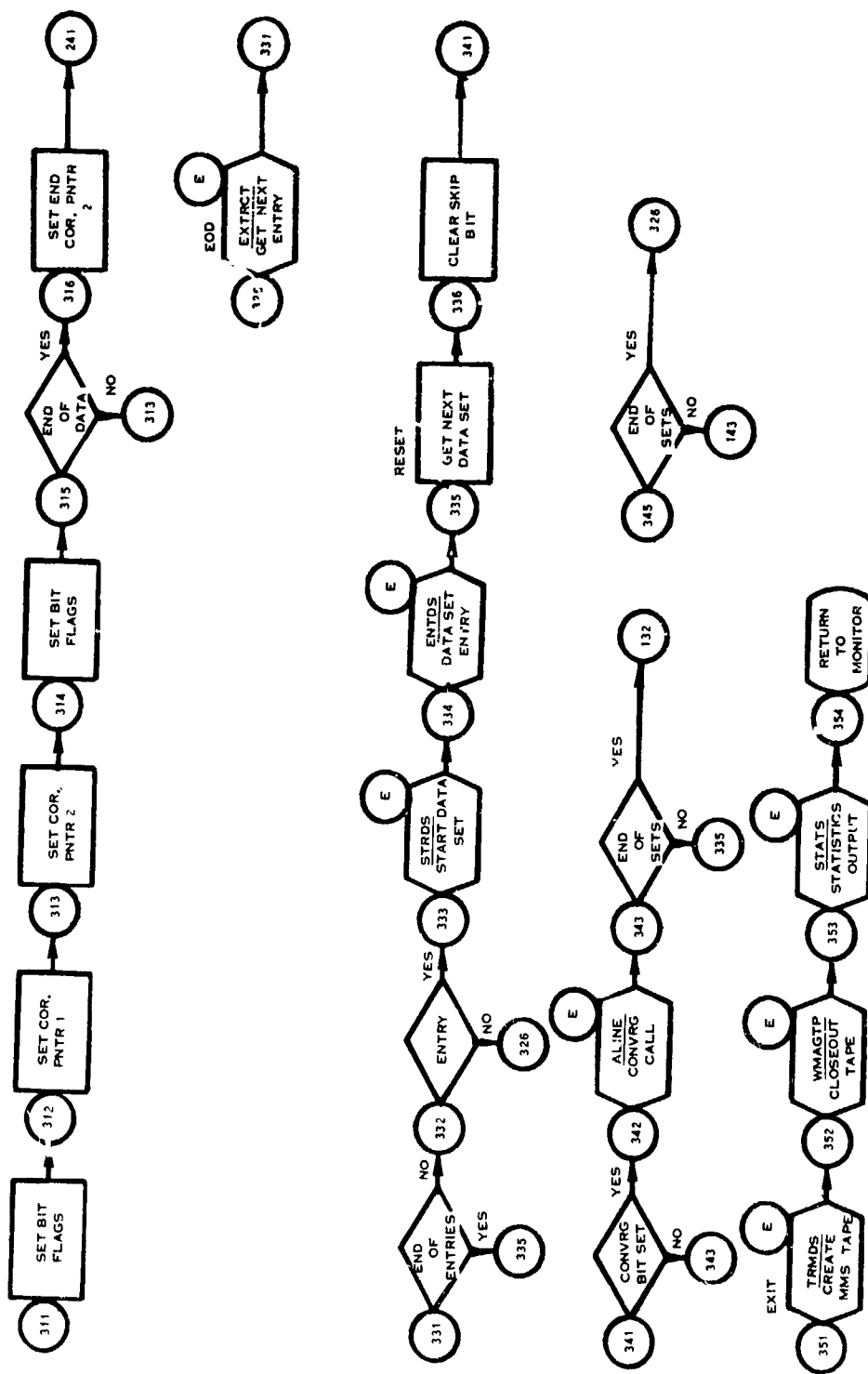


FIGURE II - 8. FLOW DIAGRAM OF CNTRL (CONTINUED)

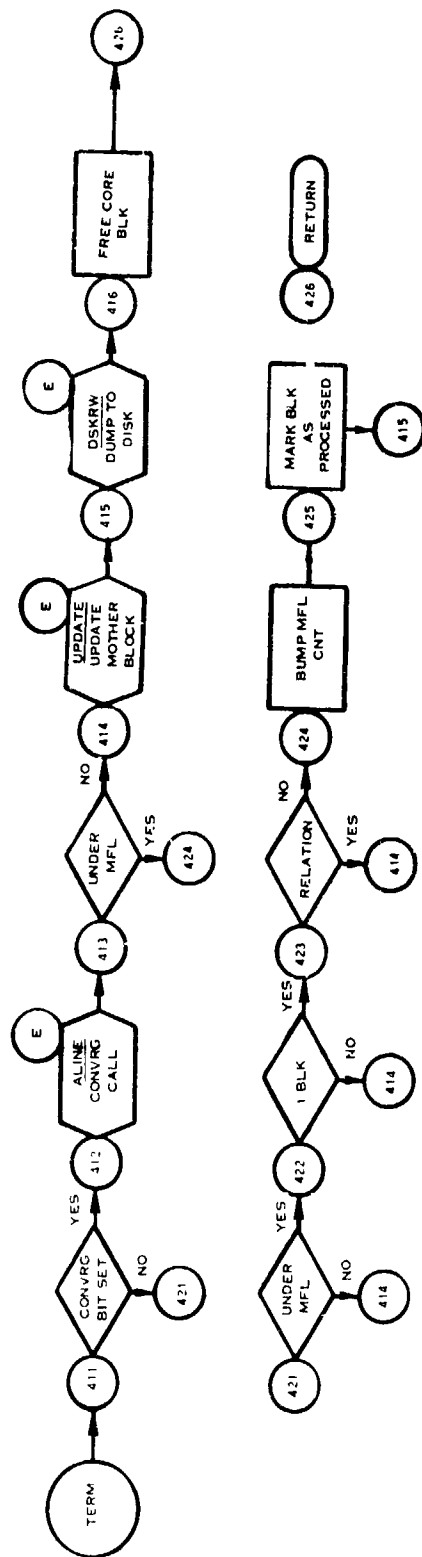


FIGURE II - 8 FLOW DIAGRAM OF CNTRL (CONCLUDED)

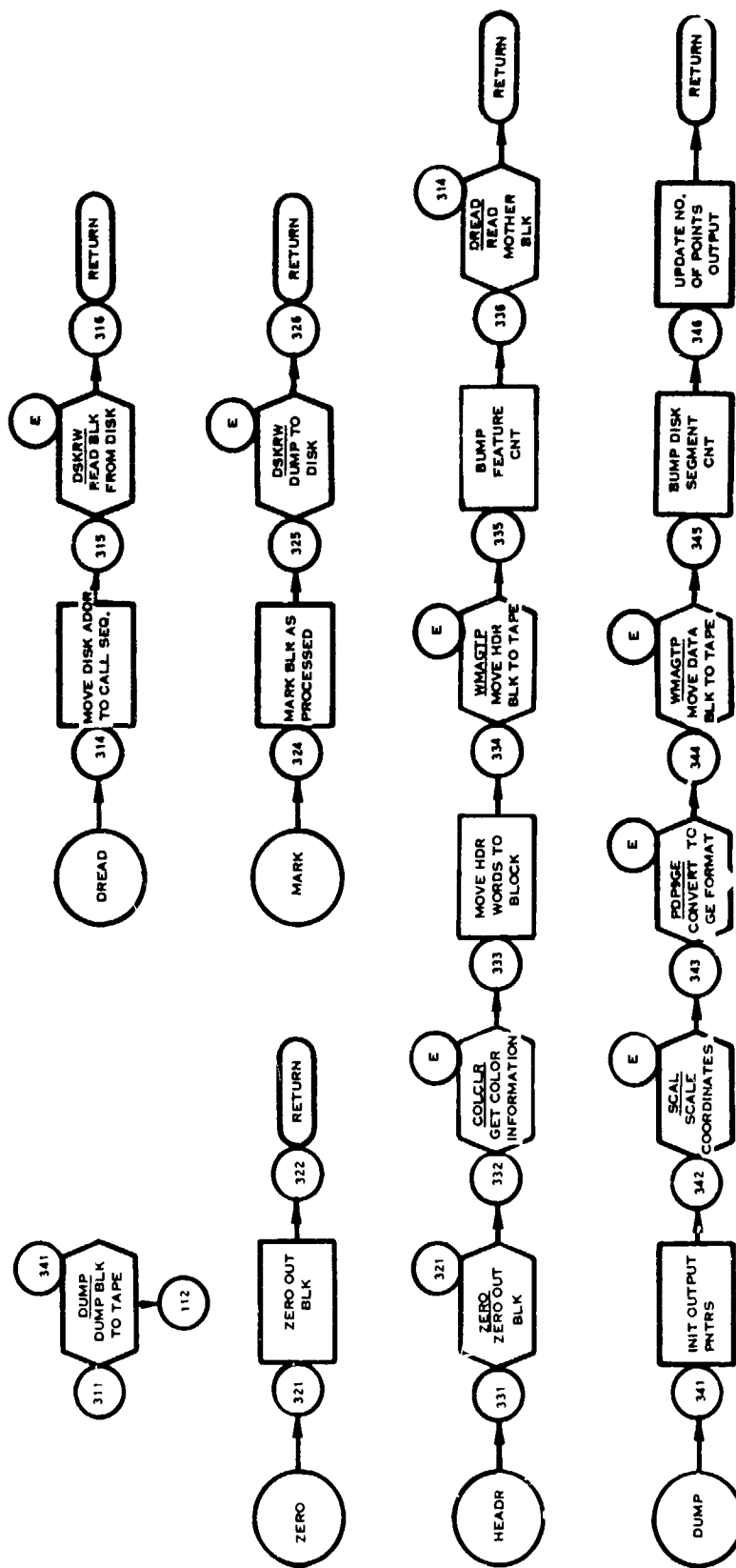


FIGURE II - 9. FLOW DIAGRAM OF TRMDS (CONCLUDED)

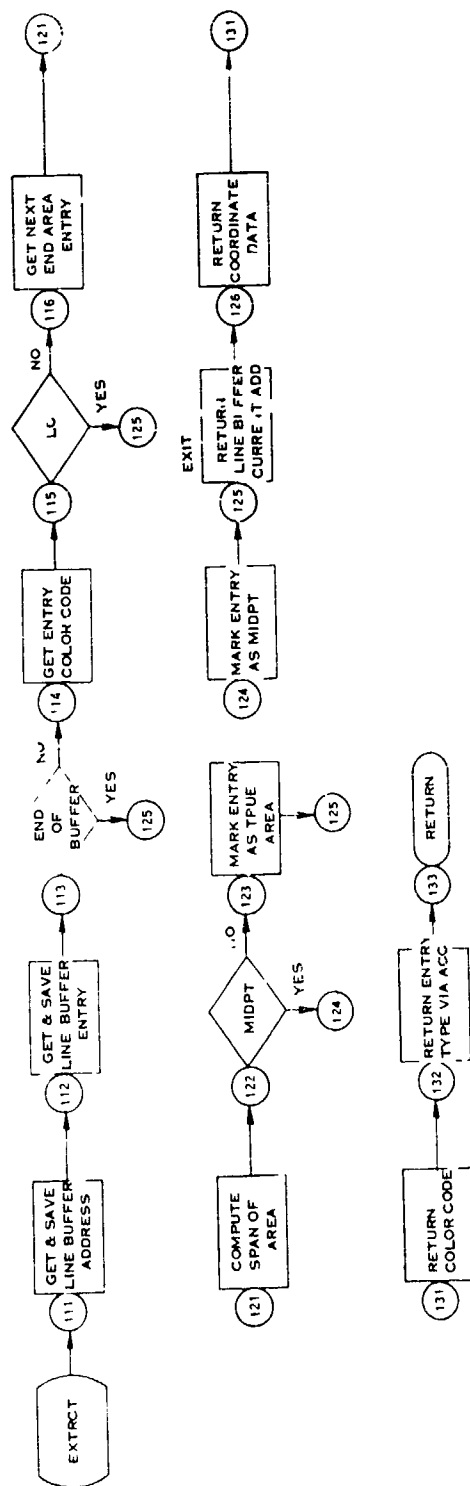


FIGURE 11 - 10. FLOW DIAGRAM OF EXTRACT

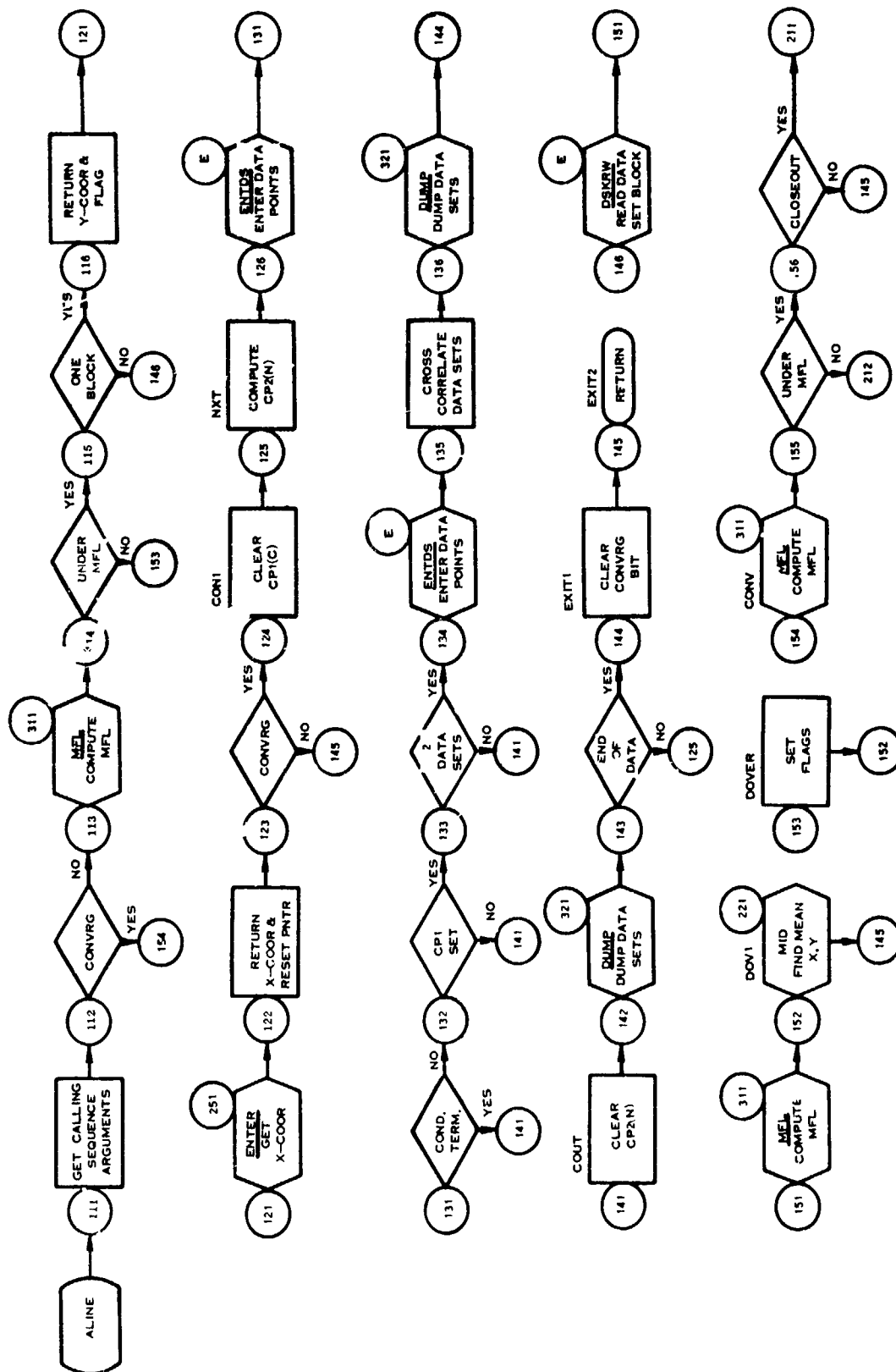


FIGURE II - 11. FLOW DIAGRAM OF ALINE

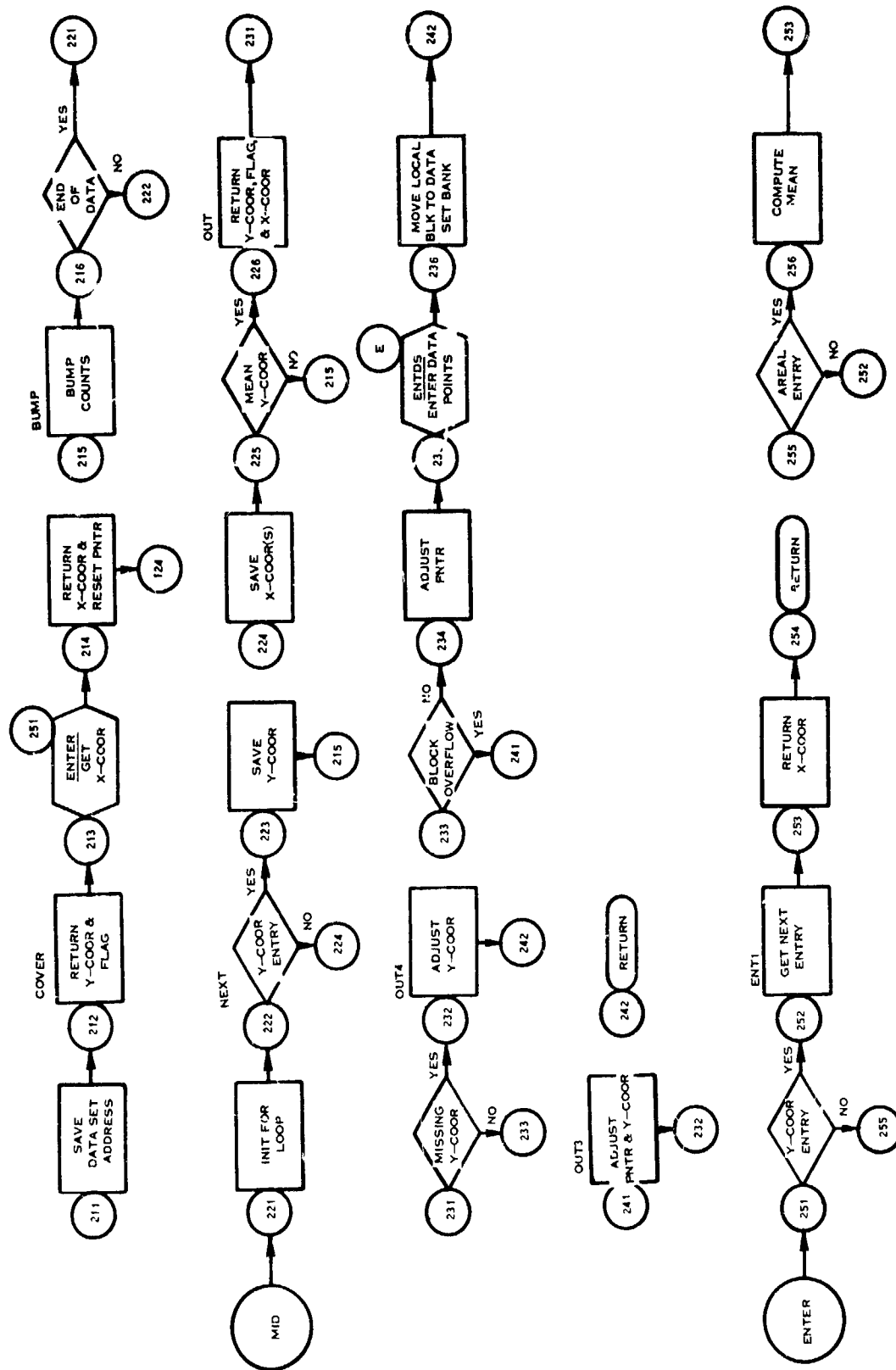


FIGURE II - 11. FLOW DIAGRAM OF ALINE (CONTINUED)

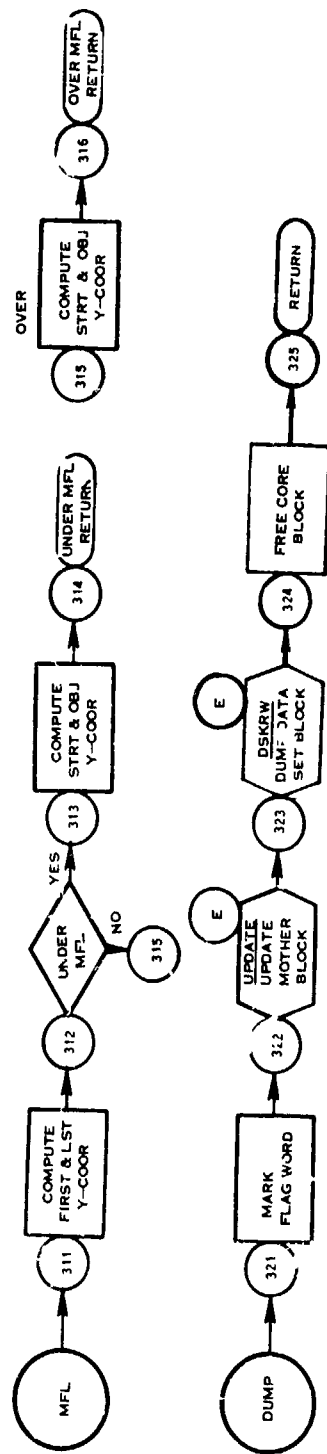


FIGURE II - il. FLOW DIAGRAM OF ALINE (CONCLUDED)

APPENDIX III

RASTER PLOTTER SOFTWARE SYSTEM OPERATING INSTRUCTIONS

APPENDIX III
TABLE OF CONTENTS

<u>Paragraph</u>		<u>Page</u>
	SECTION I	
	DISK PROGRAM CATALOGING PROCEDURE	
1.	GENERAL	121
2.	DECTAPE-TO-DISK LOADING INSTUCTIONS	121
3.	DISK-TO-CORE PDP-9 MONITOR LOADING INSTUCTIONS . . .	123
	SECTION II	
	RASTER PLOTTER SOFTWARE SYSTEM PROGRAM	
1.	CENERAL	125
2.	RPSS PROGRAM LOADING INSTRUCTIONS	125
3.	PROGRAM INITIALIZATION AND CONTROL INSTRUCTIONS . .	128
4.	TAPE MODE INITIALIZATION INSTRUCTIONS	129
5.	TAPE MODE POST-PROCESSING PRINTOUTS AND RUN TERM- INATION PROCEDURES	133
6.	MANUAL MODE INITIALIZATION INSTRUCTIONS	133
7.	POST-PROCESSING PRINTOUTS AND RUN TERMINATION PROCEDURES	139
8.	PROCESSING STATISTICAL DATA OUTPUTS	142

**APPENDIX III
INDEX OF DIAGRAMS**

<u>Figure</u>		<u>Page</u>
III-1	RPSS Program Load Map	127
III-2	RPSS Program Run (Tape Mode)	132
III-3	RPSS Program Run (Manual Mode)	140

SECTION I

DISK PROGRAM CATALOGING PROCEDURE

1. GENERAL

The following paragraphs outline in detail the procedural steps needed to build a catalog of disk-resident PDP-9 operating system and RPSS applications programs. The RPSS is designed as a stand alone program and consists of one load module of twenty-one (21) subprograms. These, in addition to the PDP-9 Operating System and Science Library reside on DECTape as object programs. In order to build an executable load module of these programs on disk storage, they must be initially loaded into core memory, and the resulting core image transferred to disk file storage. The following instructions describe this procedure.

2. DECTAPE-TO-DISK LOADING INSTRUCTIONS

a. Object Program Tape Mounting Instructions

(1) Mount object program DECTapes as follows:

- . DECTape #572 as DT 0 (PDP-9 Operating System)
- . DECTape #568 as DT 2 (RPSS Application System)

b. Load System Maintenance Routine, DSKSAV

(1) Place program, DSKSAV, in console paper tape reader

(2) Set console ADDRESS switches to 77720 (octal).

(3) Depress the following console function keys in the order indicated below:

- | | | |
|-------------|---|--|
| . STOP | - | Stop machine |
| . I/O RESET | - | Clear pending I/O and interrupt system status indicators, flags, and registers |
| . READIN | - | Initiate hard-wired "bootstrap" paper tape loading sequence. |

c. Select DECTape-to-Disk Program Loading Options

Place the following PDP-9 DISK CONTROL UNIT panel switches in the "OFF"

(down) or "ON" (up) positions at the completion of the DSKSAV loading operation. The operator will be signalled at the completion of this operation by the following system generated administrative message.

"DSKSAV"

Disk control panel settings:

- . Control switches 0-29 - "OFF"
- . WRITE PROTECT switch - "OFF"
- d. Transfer PDP-9 Operating System Object Programs from DECtape-to-Memory-to-Disk File Storage

(1) System maintenance program, DSKSAV, loads object coded executive system programs from DECtape to active core memory. The resulting core image program is then transferred onto disk. To initiate the execution of this function, perform the following sequence of console operations:

- . Set console ACCUMULATOR switches to 000000g
- . Depress CONTINUE

(2) Upon the successful completion of system program transfer from DECtape 0 to disk storage, the following administrative message is typed on-line via the console keyboard/page printer:

"DSKSAV"

- e. Reset Disk Control Panel Options to Initiate the Loading and Building of an RPSS Disk Resident Program File

(1) Place the following PDP-9 Disk Control Unit panel switches in the "OFF" (down) or "ON" (up) positions:

- . Control switches 0-29 - "ON"
- . WRITE PROTECT switch - "ON"

- f. Transfer RPSS Object Programs from DECtape-to-Memory-to-Disk File Storage

(1) Initiate transfer of RPSS object program from DECtape 2 to disk file storage as follows:

- . Set console ACCUMULATOR switches to 0000028
- . Depress CONTINUE

(2) The following message will be typed on-line when the program load and transfer operation previously described has been completed:

"DSKSAV"

3. DISK-TO-CORE PDP-9 MONITOR LOADING INSTRUCTIONS

a. Load the Disk File Resident MONITOR System Into Active Core Memory

- (1) Place system utility program, DISK, in console paper tape reader
- (2) Set console ADDRESS switches to 77637₈
- (3) Depress the following console function keys in the order specified below:

- . STOP
- . I/O RESET
- . READIN

(4) Upon the successful completion of the above program loading function, the following system generated administrative message is typed on-line:

"MONITOR V4E"

(5) The above message signals the termination of the operating and applications program system disk loading sequence, and indicates that the core resident MONITOR program is available for use.

SECTION II

RASTER PLOTTER SOFTWARE SYSTEM (RPSS) OPERATING INSTRUCTIONS

1. GENERAL

The paragraphs which follow provide the stepwise instructions and procedures for 1) loading and 2) initializing the RPSS applications program. As previously described in SECTION I, the RPSS object program is initially transferred from DEC-tape to disk file memory storage media prior to its execution by the PDP-9 hardware and executive software system. Both the program loading from disk and initialization at time of execution are performed via the on-line console teletype (ASR33). The user interacts with the memory resident PDP-9 MONITOR system to initiate and implement program link loading functions, to enter required constants, settable control parameters, and data. The on-line procedures required to enact these two functions are described separately to avoid confusion.

In the following paragraph the following notational convention is employed to differentiate between typed message outputs generated by either the MONITOR or RPSS system, and response messages entered by the user during the enactment of program loading and initialization procedures. As shown in the example below all program generated output messages are underscored, while user inputs are not.

Example:

System Output Message: MONITOR V4E

Operator Input Message: LOAD

2. RPSS PROGRAM LOADING INSTRUCTIONS

a. MONITOR V4E

System generated message to signal the operator that the system MONITOR is core memory resident and available for use (see Paragraph 3.0 SECTION I).

b. \$

System generated mnemonic indicating that the MONITOR is ready to receive operator inputs.

c. A MTF' 7

Enter above message to assign magnetic tape drive to logical unit 7.

Preceding page blank

d. LOAD

Entry of above message will cause the MONITOR to read the disk operating system link loader routine, LOAD, into active core memory.

e. LOADER V4A

System response message typed on-line after the retrieval and loading of program, LOAD, has been completed. This is followed by the system generated mnemonic;

>

which signals that, LOAD, is memory resident and available to accept user inputs.

f. A, B, C T, U

Mnemonics A, B, ..., U are symbolic names assigned to the main and sub-programs constituting the RPSS object program load modules. The above input message must be terminated by depressing the ALT MODE Key.

g. RPSS 74451

.

.

.

.

.

.CB

60053

Core memory map output in response to input message 2,g above. Map contains names and start locations of RPSS, PDP-9 Object-Time and Science Library programs and routines loaded into active memory by the link loader program, LOAD.

h. ↑S

Operating system generated mnemonic typed on-line to signal the completion of the disk-to-core memory program link loading procedure.

At this point the "virtual" core image of the RPSS and required system's library and object-time programs reside in memory.

MONITOR V4E

\$A MTF7 7

\$LOAD

LOADER V4A

>A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U

RPSS	74466
SCODE	73473
MKENT	72414
GNR	71606
SCAN	71515
SHLSRT	71271
SRCH	71136
INTWDL	71012
OUTWDL	70701
PBYTE	70555
IMESG	70227
NPAGE	70001
PRNSTT	67331
NSET	67225
SETA	67115
BLK	67067
SETM	67014
SETC	66751
BITS	66542
MGFRW	66346
BSPEOF	66161
MTF.	65001
LPA.	64316
DTA.	53416
DKA.	47056
TIME	64243
.DA	64174
BCDIO	61145
AUXIO	61047
.SS	60767
GOTO	60741
STOP	60726
PAUSE	60712
SPMSG	60617
FIOPS	60057
OTSER	46762
INTEGE	46643
REAL	45607
.CB	60037

1S1Q6

FIGURE III-1. RPSS PROGRAM LOAD MAP

j. ↑Q6

Entry of the above response input message causes the MONITOR to transfer the current core image to AREA 6 of disk file memory.

k. MONITOR V4E

Operating System generated administrative message output to signal the completion of the previously described core-to-disk memory function (see 2. j.).

l. \$

Operating System generated mnemonic output on-line to signal the "ready" status of the MONITOR; i.e., to receive user inputs.

m. G 6

Entry of the above input message cause the load module residing in disk AREA 6 to be read into active memory.

n. ↑S

Entry of the "Control S" message causes control to be transferred from the MONITOR to the entry point of the main program, RPSS. Execution of the application program will begin at this point with the initialization function. This is described in further detail in Paragraph 3.0.

3. PROGRAM INITIALIZATION AND CONTROL INSTRUCTIONS

This section provides a detailed description of the procedural steps needed to initialize the Raster Plotter Software System. The program initialization process is implemented by means of a query-response control mechanism. The on-line teletype serves as the primary communications link between man and machine, and is employed to generate administrative messages instructing the user of the specific data and control parameters needed for normal execution; to generate diagnostic messages signaling the type of keying or procedural errors committed by the user during the initialization process; and to serve as the input medium for user response messages.

All user input messages must be terminated by a carriage return, unless otherwise indicated.

The RPSS is designed to execute in one of two selectable operational modes; Automatic or "Tape Mode" and Semi-automatic or "Manual Mode".

a. Tape Mode (Automatic) -- This mode of operation processes all the raster formatted records appearing on the input data file in a serial manner, translating

each from its raw to its final raster record format. Provisions are not included to permit manual selection of data conversion based upon either channel or density parameters, or both. Instructions for initializing this mode of operation are provided in Paragraph 4.

b. Manual Mode (semi-automatic) -- This is the alternate mode of operation and provides the user with provisions for feature class selection criteria based upon various channel and density combinations. There are thirty-two (32) density/channel combinations possible, and these may be specified and assigned to known feature types at time of execution by means of the on-line Keyboard entry device. The stepwise procedures for enacting this option are described in Paragraph 6.

4. TAPE MODE INITIALIZATION INSTRUCTIONS

a. MOUNT NEW SCRATCH ON DRIVE n ENTER D WHEN DONE

Mount a full magnetic scratch tape, and set to the logical unit specified. When this task is completed respond with the following input message:

D (CR)

Note: (CR) - carriage return

b. ENTER MONTH, DAY, YEAR AS 6 NUMERICS ... MMDDYY

Enter date in the format specified by the above output message.

EXAMPLE: For February 18, 1971, enter:

021871 (CR)

c. ENTER 5 DIGIT TAPE ID NUMBER

Five digit tape identification number used to uniquely define the current output data file.

EXAMPLE

01239 (CR)

d. MOUNT FIRST REEL OF INPUT TAPE ON DRIVE m

Mount input data file and set to logical unit specified. NOTE, no response required after the completion of this operator task. However, failure to execute this step in the initialization sequence will result in a "conditional halt" when this unit is initially selected during the execution phase of RPSS.

e. ENTER 6 DIGIT MINIMUM AREA IN PROPER RESOLUTION

Entry equivalent to MIN AREA setting at which the source input graphic was processed by the ACSD. For example, if the machine setting was M. A. = 3, this would be entered as:

000003 (CR)

f. DO YOUR WISH TO SUPPLY SELECTION CODES EXTERNALLY

ENTER Y OR N

System generated message stating the option modes of operation. A message response of:

N (CR) - Indicates "TAPE MODE" selection

Y (CR) - Indicates "MANUAL MODE" selection

g. Enter N (CR)

h. ARE THERE MORE INPUT TAPES FOR THIS RUN IN ADDITION TO THAT

CURRENTLY MOUNTED ON DRIVE m ENTER Y OR N

Y (CR) denotes multi-reel input file

N (CR) denotes single-reel input file

i. MOUNT NEXT INPUT ON DRIVE i

This output message is only printed if the response to the previous query message (4 h) is affirmative (i. e., Y). No operator response is required at the completion of this task.

j. ENTER D IF PROPER INPUT TAPE MOUNTED ON DRIVE m

The above message is generated irrespective of the response to message 4. h. Its purpose is to provide an additional check point for input tape assignment validation.

Respond with the following input message for acknowledgement of input file assignment:

D (CR)

k. Entry of the above input message initiates the input data file(s) format conversion process. No further messages will appear until either the input data file is exhausted or the output is filled. In the latter case the following system generated message will appear.

1. AT PAUSE 2, MOUNT NEW SCRATCH ON DRIVE n AND CONTINUE

PAUSE 000002

When the above messages appear, remove the tape from the unit specified by numeric, n, and replace it with a full reel of magnetic tape.

To continue, enter the following input message:

↑P (CNTRL P) - Causes processing to continued from the point of interruption utilizing a new output tape file.

m. FIRST Y COORDINATE IS yyyyyy

Message to the operator indicating the Y address of the first scan line encountered so that the graphic plotter carriage address may be set at this value. This number is in decimal. No operator response is required at the completion of this message.

n. BAD WRITE AT IWRITE = wwwww

Message to the operator indicating the probability of erroneous data being written on the output tape. The scan line where this occurs is in the vicinity of the first Y coordinate plus the value of IWRITE. This number is in decimal. No operator response is required at the completion of this message.

o. BAD READ AT IREAD = rrrrr

Message to the operator indicating the probability of erroneous data being read from the input data tape. The scan line where this occurs is in the vicinity of the first Y coordinate plus the value of IREAD. This number is in decimal. No operator response is required at the completion of this message.

p. TOTAL READS = rrrrr

TOTAL WRITES = wwwww + 1

Message to the operator indicating the number of blocks of data read from the input tape and written onto the output tape. These numbers are in decimal. No operator response is required at the completion of this message.

MONITOR V4E

\$G 6

IS

MOUNT NEW SCRATCH ON DRIVE 5 ENTER D WHEN DONE

D

ENTER MONTH, DAY, YEAR AS 6 NUMERICS ... MMDDYY

072071

ENTER 5 DIGIT TAPE ID NUMBER

11111

MOUNT 1ST REEL OF INPUT TAPE ON DRIVE 6

ENTER 6 DIGIT MINIMUM AREA IN PROPER RESOLUTION.

000003

DO YOU WISH TO SUPPLY SELECTION CODES EXTERNALLY

ENTER Y OR N

N

ARE THERE MORE INPUT TAPES FOR THIS RUN IN ADDITION TO THAT
CURRENTLY MOUNTED ON DRIVE 6 ENTER Y OR N

N

ENTER D IF PROPER INPUT TAPE IS MOUNTED ON DRIVE 6

D

FIRST Y COORDINATE IS 212

TOTAL READS = 330

TOTAL WRITES = 328 + 1

DO YOU WISH TO TERMINATE RUN (Y OR N)

Y

REMOVE OUTPUT TAPE ON DRIVE 5

LABEL IT D47E 7/20/71 TAPE NUMBER 11111 REEL NUMBER 0

END OF RUN - REMOVE ALL TAPES FROM DRIVES

*** TERMINATION ***

STOP 000000

FIGURE III-2. RPS₅ PROGRAM RUN; TAPE MODE

5. TAPE MODE POST-PROCESSING PRINTOUTS AND RUN TERMINATION PROCEDURES

Upon completing an input data file processing cycle, a tabulation of run and I/O statistical data will be output on the PDP-9 line printer. These include elapsed run time, the number of input records processed, and the number output for each channel. At the conclusion of this file the following query-response dialogue is initiated.

a. DO YOU WISH TO TERMINATE RUN (Y OR N)

Y (CR) - Denotes unconditional termination of program. Control is returned to MONITOR system.

N (CR) - Denotes a request for an additional processing cycle.

b. DO YOU WISH TO STACK OUTPUT FILES (Y OR N)

The above message is output at the console ASR33, if, and only if, the response to message 5 a is a N. The responses to this message are as follows:

Y (CR) - Indicates the continued use of the current output tape file for the next processing cycle.

N (CR) - Denotes that a new output tape file is desired.

c. REMOVE OUTPUT TAPE ON DRIVE N

LABEL IT DATE mm/dd/yy, TAPE NUMBER nnnnn, REEL NUMBER rr

This message appears if the operator response to the previous query (5 b) is negative, i.e., a or if response to 5 b is affirmative i.e., a Y. Operator response to this message is not required.

d. END OF RUN - REMOVE ALL TAPES FROM DRIVE

*** TERMINATION ***

This is the final message output by the system prior to termination and transfer of control to the resident MONITOR system.

6. MANUAL MODE INITIALIZATION INSTRUCTIONS

a. MOUNT NEW SCRATCH ON DRIVE n ENTER D WHEN DONE

Mount a full magnetic scratch tape, and set to the logical unit specified. When this task is completed respond with the following input message:

D(CR)

Note: (CR = carriage return

b. ENTER MONTH, DAY, YEAR AS 6 NUMERICS ... MMDDYY

Enter date in the format specified by the above output message.

EXAMPLE: For February 18, 1971, enter:

021871 (CR)

c. ENTER 5 DIGIT TAPE ID NUMBER

Five digit tape identification number used to uniquely define the current output data file.

EXAMPLE

01239 (CR)

d. MOUNT FIRST REEL OF INPUT TAPE ON DRIVE m

Mount input data file and set to logical unit specified. NOTE, no response required after the completion of this operator task. However, failure to execute this step in the initialization sequence will result in a "conditional halt" when this unit is initially selected during the execution phase of RPSS.

e. ENTER 6 DIGIT MINIMUM AREA IN PROPER RESOLUTION

Entry equivalent to MIN AREA setting at which the source input graphic was processed by the ACSD. For example, if the machine setting was M.A. = 3, this would be entered as:

000003 (CR)

f. DO YOU WISH TO SUPPLY SELECTION CODES EXTERNALLY
ENTER Y OR N

System generated message stating the option modes of operation. A message response of:

N (CR) - Indicates "TAPE MODE" selection

Y (CR) - Indicates "MANUAL MODE" selection

Enter

Y (CR)

g. ENTER DEFAULT DENSITY AS NUMERIC FROM 0 TO 7

Enter a default density so that each output will have a density on the raster plotter. For example to have density of 3, enter
3 (CR).

h. ENTER CHANNEL NUMBER FROM 1 TO 4 OR ENTER 0 TO END
SELECTION CODE INPUT EACH TIME CH NR = IS TYPED

The first of the series of channel/data query-response message sequence permitting the input of data selection keys and corresponding output channel assignment codes. This is a preparatory message informing the operator how to respond. No operator response is required at this time.

i. CH NR =

At this time enter the channel number for which the feature associated with the selection codes are to be output. Entries are as follows:

- 1 (CR) channel 1 designated or output
- .
- .
- .
- 4 (CR) channel 4 designated as output
- 0 (CR) Indicates the end of the channel/data query response cycle

j. ENTER CODES (WYZNNNNNMMMMM)

Enter list of feature extraction keys or codes in the format indicated

W Y Z N N N N N M M M M M

The WYZNM fields have the following meanings:

W - Process control field where:

- A = All else data extraction Key
- E = Data extraction exception indicator
- N = Normal extraction code entry designator
- X = End of extraction code list indicator

Y - Range definition code, where:

- S = Denotes a single or 1:1 input code to output channel definition mapping.

R = Denotes that a set of input codes between the range specified by the NNNNN and MMMMM field inclusive, are to be mapped to the given output channel.

Z - Output density designator field

Integer value ranging from 0-7

NNNNN - Input data extraction Key (or code)

Five digit decimal number, right adjusted with leading zeroes as needed.

MMMMM - Terminal range extraction Key

Five-digit decimal number, right adjusted with leading zeros as needed

If field Y is an "S", then enter

00001

(1) For a normal single selection code of density 7 with the selection code of 39 the following entry would be made

NS70003900001 (CR)

(2) If all remaining input data is desired to be output at one specific line weight and density, the All Else data extraction key of the process control field would be entered. This elevates the need to enter all the selection codes. To initiate this function the following entry would be made

AS70000200001 (CR)

This response states that for the channel indicated in response to 6i all input data unless otherwise specified will be output with a density of 7.

(3) If, in conjunction with the All Else function, a specific feature is not to be output at all, the data extraction exception indicator of the process control field would be used. An example of this function is illustrated by the following entry.

ES60000500001 (CR)

This response says that the feature whose selection code is 5₁₀ is not to be output.

(4) To terminate the selection code entries for a particular channel, the following entry would be made:

X (CR) where = space

k. OK

Response from the system to the operator when the selection code entered has been accepted. Another selection code can then be entered after the OK response has been encountered.

l. ERRONEOUS SELECTION CODE ENTRY - WILL NOT BE PUT IN TABLE ENTRY AS RECEIVED IS

W Y Z NNNNN MMMMM

AT PAUSE 1 CONTINUE OR MANUALLY TERMINATE
PAUSE 000001

A message to the operator indicating that an error existed in the last entered selection code. The operator wishes to input selection codes then he enters.

↑P (CNTRL P)

At this time the operator will then be able to re-enter the selection code. If the operator wishes to terminate execution he should enter

↑C (CNTRL C)

and control will return to the MONITOR system.

m. SELECTION CODE TABLE FULL - LAST ENTRY

W Y Z NNNNN MMMMM

A message to the operator indicating the finish of the channel/data query response cycle due to an excess of selection codes. Processing will continue as if the channel/data query response cycle terminated normally. No operator response is required at this time.

n. SELECTION CODE SUMMARY ON PRINTER. ARE CORRECTIONS REQUIRED ENTER Y OR N

This message appears at the end of the channel/data query response cycle. That is, when the response to message 6i is zero (0). The summary on the on-line printer is in numeric order according to selection codes and shows the density and

channel on which the feature will be output. (Channels here are indicated as 0, 1, 2, 3). If there are errors, a response of

Y (CR)

will restart the channel/density query response cycle. If the table is correct, enter

N (CR).

- o. ARE THERE MORE INPUT TAPES FOR THIS RUN IN ADDITION TO THAT CURRENTLY MOUNTED ON DRIVE m ENTER Y OR N

Y (CR) denotes multi-reel input file

N (CR) denotes single reel input file

- p. MOUNT NEXT INPUT OR DRIVE i

This output message is printed if, and only if, the response to the previous query message (6.0) is affirmative (i. e., Y). No operator response is required at the completion of this task.

- q. ENTER D IF PROPER INPUT TAPE MOUNTED ON DRIVE m

The above message is generated irrespective of the response to message 6.0. Its purpose is to provide an additional check point for input tape assignment validation.

Respond with the following input message for acknowledgement of input file assignment:

D (CR)

r. Entry of the above input message initiates the input data file(s) format conversion process. No further messages will appear until either the input data file is exhausted or the output is filled. In the latter case the following system generated message will appear.

- s. AT PAUSE 2, MOUNT NEW SCRATCH ON DRIVE r AND CONTINUE PAUSE 000002

When the above messages appear, remove the tape from the unit specified by numeric, r, and replace it with a full reel of magnetic tape. To continue, enter the following input message:

↑ P (CNTRL P) - causes processing to continue from the point of interruption utilizing a new output tape file.

t. FIRST Y COORDINATE IS yyyyy

Message to the operator indicating the Y address of the first scan line encountered so that the graphic plotter carriage address may be set at this value. This number is in decimal. No operator response is required at the completion of this message.

u. BAD WRITE ATA IWRITE = wwwww

Message to the operator indicating the probability of erroneous, data being written on the output tape. The scan line where this occurs is in the vicinity of the first Y coordinate plus the value of IWRITE. This number is in decimal. No operator response is required at the completion of this message.

v. BAD READ AT IREAD = rrrrr

Message to the operator indicating the probability of erroneous data being read from the input data tape. The scan line where this occurs is in the vicinity of the first Y coordinate plus the value of IREAD. This number is in decimal. No operator response is required at the completion of this message.

w. TOTAL READS = rrrrr
TOTAL WRITES = wwwww + 1

Message to the operator indicating the number of blocks of data read from the input tape and written onto the output tape. These numbers are in decimal. No operator response is required at the completion of this message.

7. POST-PROCESSING PRINTOUTS AND RUN TERMINATION PROCEDURES

Upon completing an input data file processing cycle, a tabulation of run and I/O statistical data will be output on the PDP-9 line printer. These include elapsed run time, the number of input records processed, and the number output for each channel. At the conclusion of this file the following query-response dialogue is initiated.

a. DO YOU WISH TO TERMINATE RUN (Y OR N)

Y (CR) - Denotes unconditional termination of program. Control is returned to MONITOR system.

N (CR) - Denotes a request for an additional processing cycle.

b. DO YOU WISH TO STACK OUTPUT FILES (Y OR N)

The above message is output at the console ASR35 if, and only if, the

MONITOR V4E

\$G 6

15

MOUNT NEW SCRATCH ON DRIVE 5 ENTER D WHEN DONE

D

ENTER MONTH, DAY, YEAR AS 6 NUMERICS ... MMDDYY

072071

ENTER 5 DIGIT TAPE ID NUMBER

99999

MOUNT 1ST REEL OF INPUT TAPE ON DRIVE 6

ENTER 6 DIGIT MINIMUM AREA IN PROPER RESOLUTION.

000003

DO YOU WISH TO SUPPLY SELECTION CODES EXTERNALLY

ENTER Y OR N

Y

ENTER DEFAULT DENSITY AS NUMERIC FROM 0 TO 7

6

ENTER CHANNEL NUMBER FROM 1 TO 4 OR ENTER 0 TO END

SELECTION CODE INPUT EACH TIME CH NR = IS TYPED

CH NR =

4

ENTER CODES (WYZNNNNNNMMMM)

NS50004700001

OK

X

CH NR =

0

SELECTION CODE SUMMARY ON PRINTER. ARE CORRECTIONS REQUIRED

ENTER Y OR N.

N

FIGURE III-3. RPSS PROGRAM RUN:MANUAL MODE

ARE THERE MORE INPUT TAPES FOR THIS RUN IN ADDITION TO THAT
CURRENTLY MOUNTED ON DRIVE 6 ENTER Y OR N

N

ENTER D IF PROPER INPUT TAPE IS MOUNTED ON DRIVE 6

D

FIRST Y COORDINATE IS 212
TOTAL READS = 330
TOTAL WRITES = 328 + 1
DO YOU WISH TO TERMINATE RUN (Y OR N)

Y

REMOVE OUTPUT TAPE ON DRIVE 5
LABEL IT DATE 7/20/71 TAPE NUMBER 099999 REEL NUMBER 0
END OF RUN - REMOVE ALL TAPES FROM DRIVES

. * TERMINATION ***

STOP 000000

FIGURE III-3. RPSS PROGRAM RUN:MANUAL MODE (CONCLUDED)

response to message 7a is a N. The response to this message are as follows:

Y (CR) - Indicates the continued use of the current output tape file for the next processing cycle.

N (CR) - Denotes that a new output tape file is desired.

- c. REMOVE OUTPUT TAPE ON DRIVE n
LABEL IT DATE mm/dd/yy, TAPE NUMBER rrrrrr, REEL NUMBER rr

This message appears if the operator response to the previous query (7b) is negative, i.e., an or if response to 7a is affirmative i.e., a Y. Operator response to this message is not required.

- d. DO YOU WISH SAME SELECTION CODES AS PREVIOUS RUN (Y OR N)

If additional processing cycle is requested (i.e., response to 7a is N), this message appears to determine if the selection codes for the new input tape are the same as the previous. If they are, then the channel/data query response cycle is avoided, in this case the response should be

Y (CR).

If there are any changes in the selection codes, the channel/data query response cycle must be restarted and in this case, the response should be

N (CR).

This latter response can also be used to change mode of operation to Tape Mode, if so desired.

- e. END OF RUN - REMOVE ALL TAPES FROM DRIVES

TERMINATION

This is the final message output by the system prior to termination and transfer of control to the resident MONITOR system.

8. PROCESSING STATISTICAL DATA OUTPUTS

Immediately following the successful completion of the generation of a drive tape for the graphic plotter, timing and data statistics are presented on-line via the line printer. These include the number of scan lines input and output, number of data points processed, the number of points of various ACSD designator codes and the number of points output per channel.

APPENDIX IV

RASTER PLOTTER SOFTWARE SYSTEM PROGRAM DESCRIPTION

APPENDIX IV

TABLE OF CONTENTS

<u>Paragraph</u>		<u>Page</u>
	SECTION I	
	RASTER PLOTTER SOFTWARE SYSTEM PROGRAM	
1	GENERAL	143
2	SYSTEM DESCRIPTION	143
a.	Object Program Deck Arrangement	143
b.	System Diagrams	144
c.	Logic Diagram Symbolology	144
	SECTION II	
	DESCRIPTION OF SUBPROGRAMS	
1	WORKER AND UTILITY PROGRAMS	153
a.	RPSS (Raster Plotter Software System Control Program)	153
b.	SCODE (Selection Code)	156
c.	MKENT (Make Table Entry)	158
d.	GNR (Get Next Record)	160
e.	SCAN (Scan Selection Code Table)	161
f.	SHLSRT (Shell Sort)	162
g.	SRCH (Search The Table)	162
h.	INTWDL (Input-Table Word Logic)	163
i.	OUTWDL (Output-Table Word Logic)	165
j.	PBYTE (Byte Placement)	165

TABLE OF CONTENTS (CONT)

<u>Paragraph</u>		<u>Page</u>
k.	IMSEG (Messages)	167
l.	NPAGE (New Page)	168
m.	PRNSTT (Print Run Statistics)	168
n.	NSET (Set Buffer)	170
o.	SETA (Set Area)	171
p.	SETM (Set Masks)	171
q.	SFTC (Set Constants)	172
r.	BITS (Bit Routines)	173
s.	MGFRW (Magnetic Tape Read/Write)	175
2	OBJECT-TIME SYSTEM AND SCIENCE LIBRARY ROUTINES	178
a.	Object-Time System Routines	178
b.	Science Library Routines	179

SECTION III ASSEMBLY LANGUAGE SUBPROGRAM FLOW DIAGRAMS

SECTION IV FORTRAN SUBPROGRAM AUTOFLOW DIAGRAMS

APPENDIX IV
INDEX OF DIAGRAMS

<u>Figure</u>		<u>Page</u>
IV-1	System Logic Flow Diagram	145
IV-2	Raster Plotter Software System Block Diagram	146
IV-3	RPSS Tape Mode Block Diagram	149
IV-4	RPSS Manual Mode Block Diagram	150
IV-5	Logic Diagram Symbolology	151
IV-6	Flow Diagram of SETM	182
IV-7	Flow Diagram of SETC	182
IV-8	Flow Diagram of BITS	183
IV-9	Flow Diagram of MGFRW	184

SECTION I

RASTER PLOTTER SOFTWARE SYSTEM PROGRAM

1. GENERAL

The RPSS Program described in this appendix consists of fifteen FORTRAN language subprograms and four MACRO assembler language routines which are functionally unique.

Section II provides a detailed description of the nineteen unique subprograms of the RPSS Program. Subsection 1 contains the program descriptions of both the FORTRAN and MACRO subprograms. Subsection 2 identifies the existing PDP-9 Object-Time System programs and Science Library MACRO subroutines called by the RPSS Program for various FORTRAN IV input/output functions. Section III contains the flow diagrams for the required RPSS assembly language subprograms. Section IV contains the autflow diagrams for the required RPSS FORTRAN subprograms.

2. SYSTEM DESCRIPTION

The RPSS Program is a self-contained, stand alone program designed to be executed on a PDP-9 processing system having the following configuration:

<u>Quantity</u>	<u>Equipment Item Description</u>
1	DEC PDP-9 Control Processing Unit (CPU)
8	4096 18-bit Word Core Memory Modules
1	10 CPS, Console Keyboard/Teleprinter
3	7-Track, 45IPS, 556/800 BPI IBM Compatible Magnetic Tape Units W/Controller
4	DECtape Units W/Controller
1	DEC PDP-9 Paper Tape Reader
1	DEC PDP-9 Paper Tape Punch
1	1 Million 18-Bit Word Disk File W/Controller
1	ANELEX, 1000 LPM Line Printer

a. Object Program Deck Arrangement

The RPSS Program is assembled and compiled as a "relocatable" program. With the exception of the control program (RPSS), which by definition must be the first program in the object deck, the remaining object programs may be arranged in any order.

Preceding page blank

b. System Diagrams

Figure IV-1 shows the control logic flow for the RPSS program. PDP-9 OTS and Science Library subprograms and routines are not shown since they are loaded and enter the procedural flow only at the time of execution.

Figure IV-2 shows the overall system block diagram of the RPSS program, while Figure IV-3 and IV-4 show the Tape and Manual mode block diagrams respectively.

c. Logic Diagram Symbolology

The syllabus of logic diagram symbolology, as shown in Figure IV-5, defines the program flow chart notation conventions employed in this appendix.

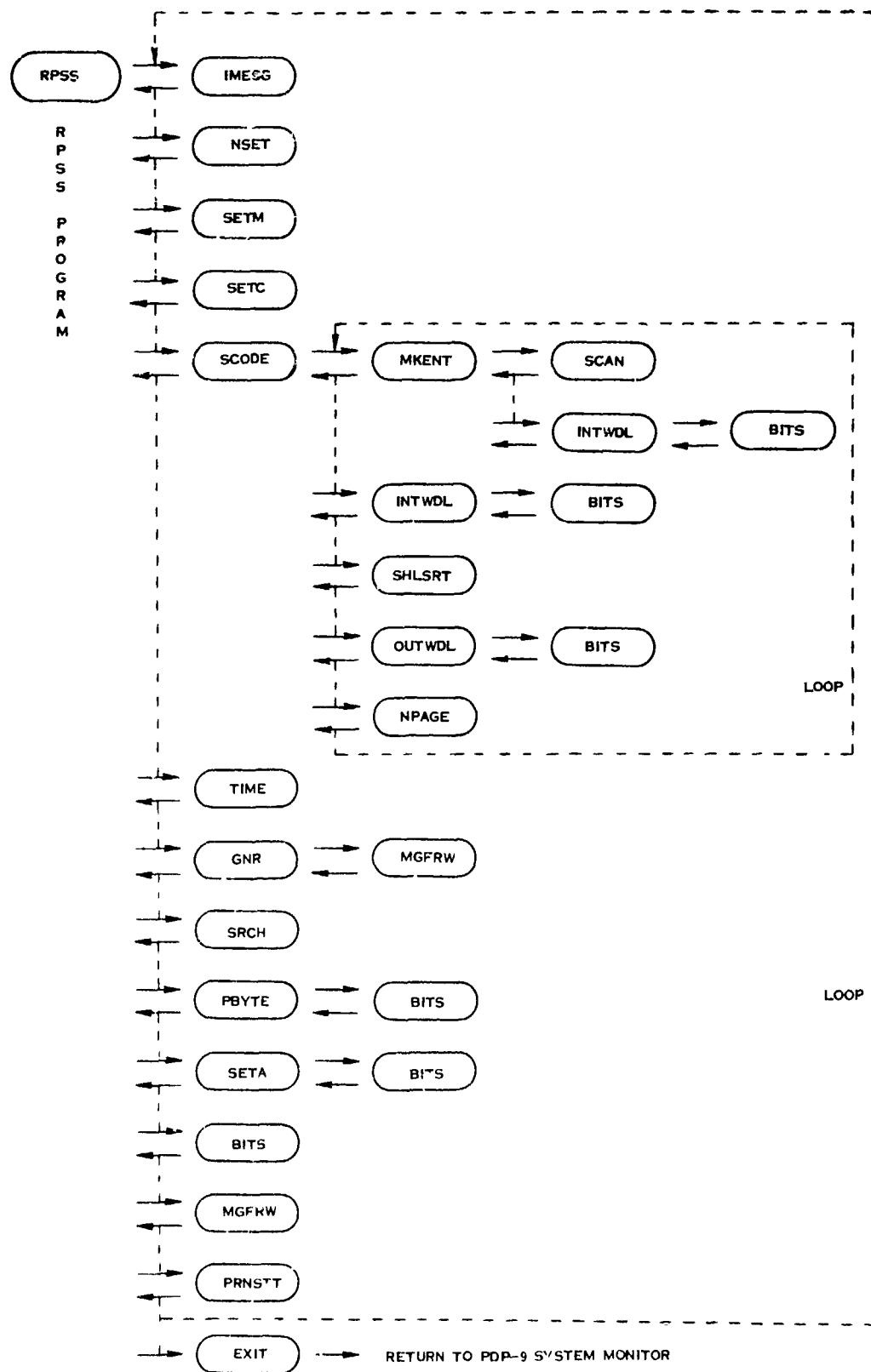


FIGURE IV-1. SYSTEM LOGIC FLOW DIAGRAM



FIGURE IV - 2. RASTER PLOTTER SOFTWARE SYSTEM BLOCK DIAGRAM (CONTINUED)

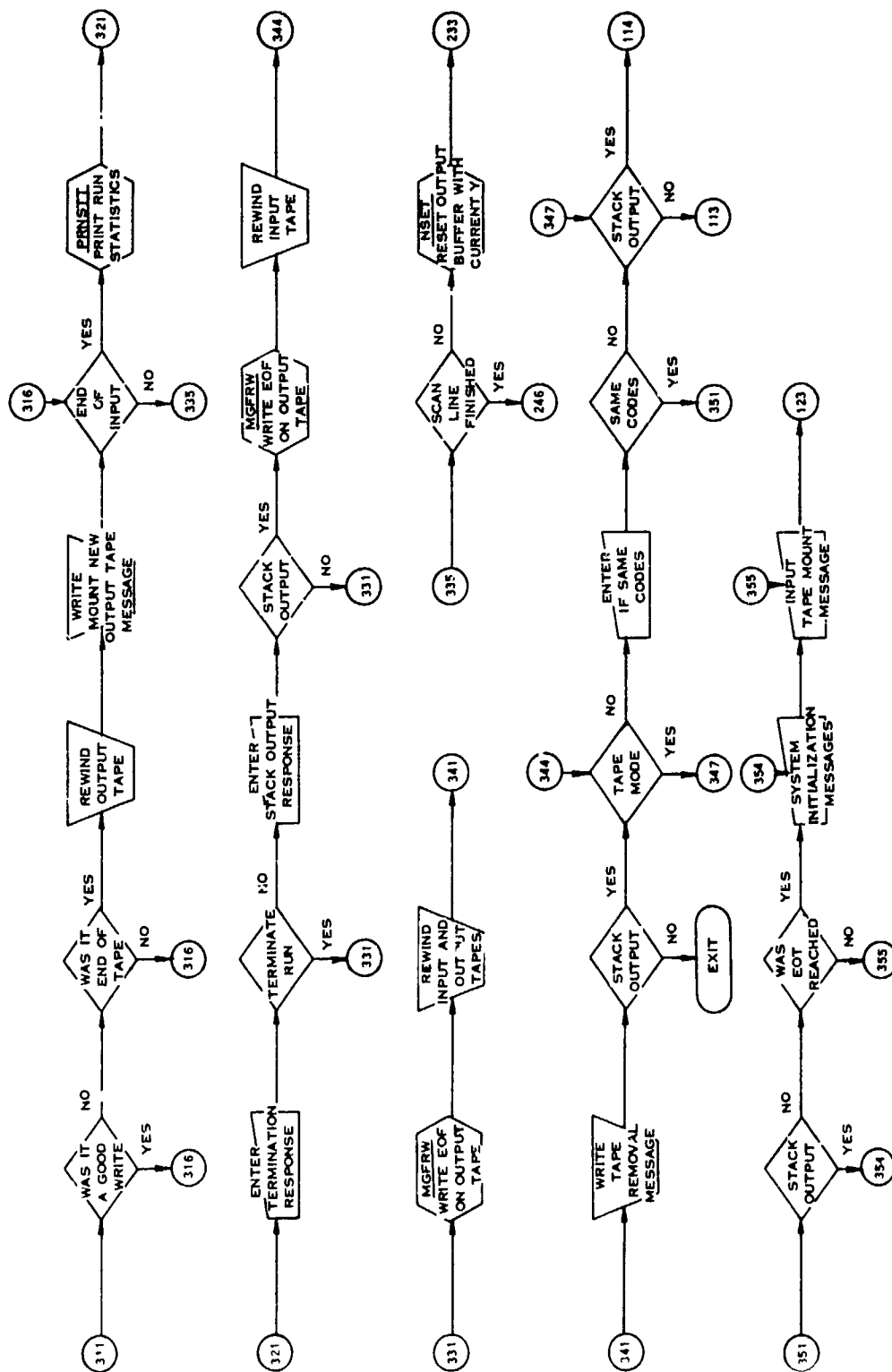


FIGURE IV - 2. RASTER PLOTTER SOFTWARE SYSTEM BLOCK DIAGRAM (CONCLUDED)

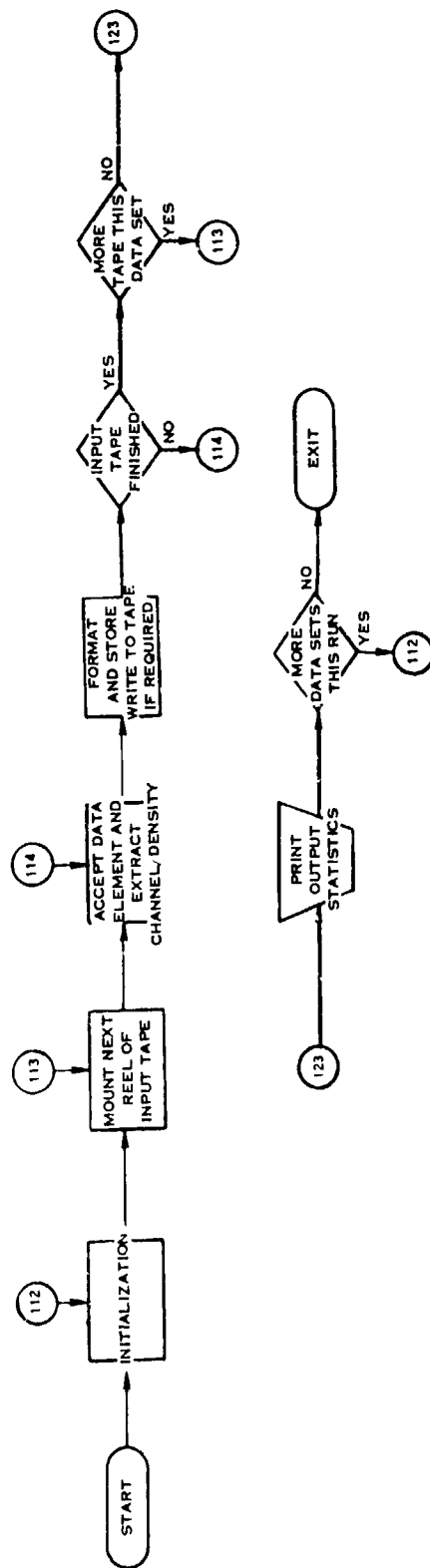


FIGURE IV - 3. RPSS TAPE MODE BLOCK DIAGRAM

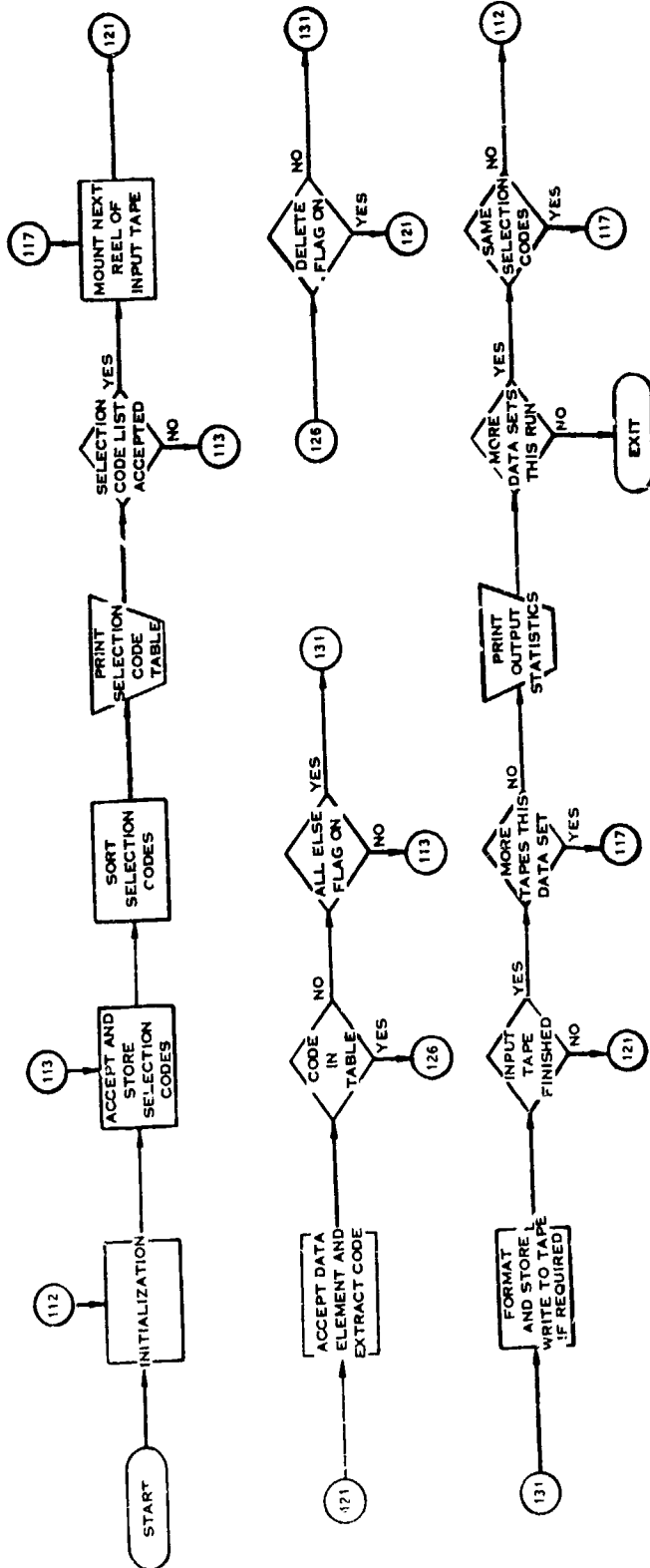
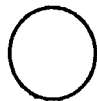


FIGURE IV - 4. RPSS MANUAL MODE BLOCK DIAGRAM



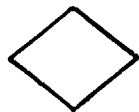
PROGRAM NAME BOX



SUBPROGRAM NAME BOX



FUNCTION DESCRIPTION BOX



DECISION BOX



INTERNAL SUBPROGRAM CALL BOX
X--FLOW DIAGRAM PAGE NUMBER (X OF N)
Y--LINE ON PAGE
Z--BOX IN LINE



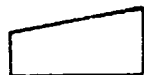
EXTERNAL SUBPROGRAM CALL BOX



PDP-9 SYSTEM CALL BOX



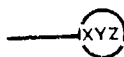
HARDCOPY OUTPUT



TELETYPE/TELEPRINTER (TTY)



INPUT/OUTPUT BOX



LOGICAL P
X--FLOW DIAGRAM PAGE
Y--LINE ON PAGE
Z--BOX IN LINE

FIGURE IV-5. LOGIC DIAGRAM SYMBOLOGY

SECTION II

DESCRIPTION OF SUBPROGRAMS

1. WORKER AND UTILITY PROGRAMS

a. RPSS (Raster Plotter Software System Control Program)

(1) Abstract

RPSS is the main control program of the Raster Plotter Software System. It oversees all the supervisory functions required to handle the output format and determines whether the tape format or the selection code assignment format is to be utilized. The program then reads raster scan lines from the input tape by calling GNR, locates and extracts the proper selection code using the selection code table (SRCH) and reformats and places it into the output buffer area. The program then calls MGFRW to write the record on the output tape. Upon completion, statistical information is output by PRNSTT via the on-line printer and control is returned to the PDP-9 monitor system.

(2) Program Description

RPSS, during system initialization, receives via the on-line teletype all the pertinent information necessary to process the selected raster format input tape. Formatting and selection for output can proceed in one of two paths. The first path is that of data transfer and reorganization for the final output tape. In this mode, referred to as Tape Mode, it is necessary for the user to supply a minimum area for future recognition of this parameter in the data set. The input information is then read from the input tape by a call to GNR. If no EOF or EOT is encountered, the data point is typed, i. e., whether it is an X - or Y - coordinate and is then placed, after formatting, in an output buffer, located in core, for future transfer to tape. The coordinate is stored with the associated line center designator and/or area start/stop, channel number, and density in proper plotter format. For the minimum area case an additional area stop coordinate is generated. The buffer is written to the output tape when one of two conditions occur. The first when the buffer is filled and the second is when a new Y - coordinate, is encountered before the buffer is filled. Tape reading and writing is handled by calls to MGFRW. The second mode of operation, referred to as the Manual Mode, allows the user to select which features on the input tape he would like to have on the output tape. Each feature has a unique selection code for ease in recognition. SCODE handles all the necessary procedures to obtain the required information for selection code, channel number and density. The procedure for data handling is the same then as for the tape mode with the addition that each coordinate has its "selection code" compared to the selection code table (by SRCH) to ensure that it is an extracted feature. If it is not a desired feature, it is not placed in the output buffer; if it is a desired feature then all pertinent information for

proper plotter operation is placed in the buffer. Processing then continues as previously discussed. Statistical information is compiled throughout execution and is output via the on-line printer at the end of each run upon completion of successful execution.

(3) Inputs

(a) Calling Sequence -

None - stand alone program.

(b) Keyboard Entries - Operator Input Responses -

D

mmddyy (month, day, year)

iiiii (tape ID)

Y

N

aaaaaa (minimum area)

CNTL P (for continuation)

(c) Magnetic Tape -

Raster formatted data tape containing all necessary information in proper format.

(4) Outputs

(a) Teleprinter -

Program generated query and administrative messages

"MOUNT NEW SCRATCH ON DRIVE ENTER D WHEN DONE"

"ENTER MONTH, DAY, YEAR AS 6 NUMERICS . . .MMDDYY"

"ENTER 5 DIGIT, TAPE ID NUMBER"

"MOUNT 1ST REEL OF INPUT ON DRIVE _____"

"DO YOU WISH TO SUPPLY SELECTION CODES EXTERNALLY ENTER Y OR N"

"ENTER 6 DIGIT MINIMUM AREA IN PROPER RESOLUTION"

"AT PAUSE 2, MOUNT NEW SCRATCH ON DRIVE _____ AND CONTINUE"

"DO YOU WISH TO TERMINATE RUN? (Y OR N)"

"REMOVE OUTPUT TAPE ON DRIVE _____ LABEL IT: DATE; _____/_____/_____, TAPE NUMBER; _____ REEL NUMBER _____"

"DO YOU WISH TO USE SAME SELECTION CODES AS PREVIOUS RUN (Y OR N)"

"END OF RUN - REMOVE ALL TAPES FROM DRIVES"

*** TERMINATION ***

"BAD WRITE AT IWRITE = _____"

"TOTAL WRITE S = _____ + 1"

(b) Magnetic Tape -

Raster formatted tape for use on the Graphic Plotter.

(5) Referenced Subprograms

SCODE
GNR
SRCH
PBYTE
IMESG
PRNSTT
NSET
SETA
SETM
SETC
MGFRW

Bit Routines:	LS
	LO
	LEOR
	LA
	RS

b. SCODE (Selection Code)

(1) Abstract

SCODE is that part of the Raster Plotter Software System that handles the selection code processing before reading the input data tape. This sub-routine accepts the selection codes, densities, and channel numbers that will be used as the selection criteria for the final output product. SCODE performs all the preliminary handling of the selection code information. In addition, SCODE sets the ALL ELSE channel if and when it is selected.

(2) Program Description

SCODE reads in from the TTY Keyboard the necessary information to form the Selection Code Table (ISC). The first thing read in is the default density. The table is then preset with the default density being placed in each entry of the tables. Right after this is done the ALL ELSE functions are cleared and the ALL ELSE channel is set to channel 5. The selection code is read in on a per channel basis. Then the desired channel number is read. The procedure of obtaining the selection codes is continued until either the user signifies end of selection code assignment or until the selection code table has been completely filled. Entries to the table are made by MKENT. After the last entry has been made the Selection Code Table is sorted (SHLSRT) and the entire table in sorted order is printed via the on-line printer so that the user will have a hard copy of the selection code, channel, density, and delete status. The ALL ELSE information is also output at this time. The user is then queried to determine whether or not the list is acceptable. If not acceptable the selection code assignment procedure is repeated. If accepted, control returns to the main program to continue processing.

(3) Inputs

(a) Calling Sequence -	CALL SCODE
------------------------	------------

Arguments:	None
------------	------

(b) Teletype Keyboard -

d	(default density)
n	(channel number)
A	
R	
S	
N	
E	
X	
Y	
N	

(4) Outputs

(a) Teleprinter -

Program generated query and administrative messages.

"ENTER DEFAULT DENSITY AS NUMERIC FROM 0 TO 7"

"ENTER CHANNEL NUMBER FROM 1 TO 4 OR ENTER 0
TO END SELECTION CODE INPUT EACH TIME CH NR =
IS TYPED"

"CH NR = "

"ENTER CODES (WYZNNNNNNMMMMM)"

"SELECTION CODE SUMMARY ON PRINTER. ARE COR-
RECTIONS REQUIRED ENTER Y OR N."

(b) On-Line Printer -

Information printed in addition to that printed by subroutine
NPAGE are the selection code information to appear in tabular form. The last entry
at the end of the list would be

"ALL ELSE"

under the selection code heading, with the proper channel number and density appear-
ing under the appropriate column headings. The ALL ELSE information will be printed
only if ALL ELSE has been specified.

(5) Referenced Subprograms

INTWDL
MKENT
SHRSRT
NPAGE

c. MKENT (Make Table Entry)

(1) Abstract

MKENT reads the input selection code and enters it into the selection code table (ISC) for the selection code mode of operation. The status of the selection code table is checked before and after each entry. If there is an entry for a particular selection code already in the table, the previous entry will be overwritten. A return parameter indicates the status of the table after the entry has been made.

(2) Program Description

Before entering the selection code for a specific previously designated channel into the table, a series of error checks are made. These checks include

- a. a test for end of input
- b. erroneous entry
- c. is it the ALL ELSE channel

After these checks, MKENT determines if the delete flag is either on or off, if the default density is to be implemented or if a density has been input, and whether the selection code being input is a single value or a range of values. At this time placement into the table can proceed. Before the actual entry is made the selection code table is checked for a duplicate entry (same selection code) by a call to SCAN. If SCAN indicates a duplicate, the previous entry is overwritten. If there is no duplicate entry, the pointer is advanced to the next available spot. INTWDL is then called to place the selection code in the first word and to pack the density channel number previously designated and delete flag (if on) into the second word of the table entry. After each entry, the table is checked to see if it is full. The table is large enough to hold 1000 selection codes with associated channel number, density and delete flag. If the table is full, further inputs are ignored and processing continues without the additional inputs. If a range of selection codes was input, entries are made for each selection in the entire range until the range is exhausted. The return parameter indicates the status of the table.

(3) Inputs

(a) Calling Sequence - CALL MKENT (IFLG)

Arguments: IFLG - return parameter from MKENT telling the status of the selection code table

- 3 table is full after entry made
 - 2 erroneous entry, not processed
 - 1 end of input encountered
 - 0 All Else Channel assigned
- positive number
the location (pointer)
number assigned the entry in selection code table.

(b) Entries - Selection Code Information -

WYZNNNNNNMMMMM

W =	{	A	ALL ELSE channel assignment
		E	Except = Delete Flag Set
		N	Normal Entry
		X	End of Input
Y =	{	S	Single selection code
		R	Range of selection codes
Z =			Density
NNNNN =			Start of selection code or signal selection code
MMMMM =			End of selection code (if Y = R, otherwise 00001)

(4) Outputs

(a) Teleprinter -

"SELECTION CODE TABLE FULL - - LAST ENTRY -----"

"ERRONEOUS SELECTION CODE ENTRY - WILL NOT BE
PUT IN TABLE
ENTRY AS RECEIVED IS:

WYZNNNNNNMMMMM

AT PAUSE 1, CONTINUE OR MANUALLY TERMINATE.

(5) Referenced Subprograms

SCAN
INTWDL

d. GNR (Get Next Record)

(1) Abstract

GNR gets the next record from the input tape containing the data to be processed.

(2) Program Description

GNR issues the call to MGRW to read the input tape and store it in a local buffer. GNR then investigates the block of data. If an EOF or EOT is encountered, control is returned to the main program noting the condition. If no EOF or EOT is encountered, GNR then determines if the block is the first block for a given raster line, or a continuation block. GNR also determines whether the data is an X - coordinate or a Y - coordinate. Subsequent calls return the next coordinate from the buffer until it is empty.

(3) Inputs

(a) Calling Sequence - CALL GNR

Arguments: NONE

(4) Outputs

(a) Memory -

Buffer area filled with the record read from the input tape.

(b) Teletype -

"FIRST Y COORDINATE IS _____"

"BAD READ AT IREAD = _____"

"TOTAL READS = _____"

(5) Referenced Subprograms

MGFRW

e. SCAN (Scan Selection Code Table)

(1) Abstract

SCAN checks the Selection Code Table (ISC) before entries are made to insure that no duplicate entries exist. If a duplicate is found, its position is returned to the calling program.

(2) Program Description

SCAN is a loop that checks each member of the selection code table (ISC) against the new selection code to be entered (IVAL). This is performed on the unsorted table as the table is being built, to avoid duplicate entries. It is performed each and every time an entry is to be made in the selection code table. If a duplicate is found, a pointer to that duplicate entry is returned to the calling program in the parameter K. If no duplicate is found, that is, if the selection code has not been previously entered in the table, a return code of -1 is passed in the parameter K.

(3) Inputs

(a) Calling Sequence - CALL SCAN (IVAL, K)

Arguments:

IVAL = Selection Code for which the table is to be scanned to determine if it is a duplicate entry

K = Return argument telling status of IVAL in the selection code table where:

-1 IVAL did not previously appear in the table

≠ -1 IVAL did previously appear in the table and K contains its location with the table

(4) Outputs

None

(5) Referenced Subprograms

None

f. SHLSRT (Shell Sort)

(1) Abstract

SHLSRT performs a Shell sort on the entries of the Selection Code Table (ISC). The table is ordered in ascending selection code with each two word entry of the table being moved as necessary.

(2) Program Description

SHLSRT is an algebraic comparison of the entries in the selection code table (ISC) made on the first word of the table entry. This routine is called after the last entry has been made. The table is ordered by ascending selection code with the second word of the entry being switched when necessary, to remain with its proper selection code.

(3) Inputs

(a) Calling sequence - CALL SHLSRT

Arguments: None

(4) Outputs

None

(5) Referenced Subprograms

None

g. SRCH (Search The Table)

(1) Abstract

SRCH performs a binary search of the sorted Selection Code Table to determine if a particular selection code is in the table. The return argument

tells the status of the selection code.

(2) Program Description

SRCH is called after the input data tape is read to determine if a particular selection code is in the Selection Code Table (ISC). A binary search is incorporated to optimize the search time as the table lookup has to be done for each coordinate from the input tape. If the selection code is found a return argument indicating the location in the table is returned. If the selection code is not found, i. e., not in the table, a value of -1 is returned.

(3) Inputs

(a) Calling Sequence - CALL SRCH (IARG, K)

Arguments:

IARG = The selection code the table is being searched for.

K = Return argument. If it is -1, the selection code was not found. If a positive integer then it is a pointer to the position in the Selection Code Table to where the selection code is.

(4) Outputs

None

(5) Referenced Subprograms

None

h. INTWDL (Input - Table Word Logic)

(1) Abstract

INTWDL makes the entry in the Selection Code Table (ISC). The first word of the entry is the selection code and the second word contains the density, channel number and delete flag, packed in proper format for output on the graphic plotter.

(2) Program Description

INTWDL utilizes the bit manipulation routines in order to pack the second word of the Selection Code Table (ISC) entry. The parameters to be put into the table are paired in the labelled common B1 and the location is passed in the argument K. The selection code is immediately placed into the first word as an integer. The form of the second word is as follows:

bits	0	-	5	are zero
	0	-	8	contain the density
	9			contains a 1
	10	-	11	contains the channel number
	12	-	16	are zero
	17			is the delete flag

The second byte (bits 6 - 11) of the second word contains the density and channel number in the proper format for output on the graphic plotter. Before the entry is made, the location in the selection code table is cleared to zero to avoid any extraneous information for the bit routines.

(3) Inputs

(a) Calling Sequence- CALL INTWDL (K)

Arguments:

K = Selection Code Table location for which the entry is to be made.

(4) Outputs

(a) Memory -

Selection code table entry containing selection code, density, channel number and delete flag.

(5) Referenced Subprograms

Bit Routines: LS
 LO
 LTCOR

i. OUTWDL (Output - Table Word Logic)

(1) Abstract

OUTWDL reads the entry from the selection code table and decodes the second word of the entry into the density, channel number and delete flag.

(2) Program Description

The first word of the Selection Code Table (ISC) entry contains only the selection code. The second word in packed form contains the density, channel number and delete flag (see INTWDL). The bit manipulation routines are used to unpack the second word and extract the desired information. The unpacked parameters are then passed by the labeled common H1.

(3) Inputs

(a) Calling Sequence - CALL OUTWDL (K)

Argument:

K = Selection Code Table location for which the entry is to be extracted.

(4) Outputs

(a) Memory -

The selection code, density, channel number and delete flag stored in labeled common H1.

(5) Reference Subprograms

Bit Routines:

LA
RS

j. PBYTE (Byte Placement)

(i) Abstract

PBYTE is a byte manipulation routine that shifts bytes either left or right in a PDP-9, 18-bit (3-byte) word.

(2) Program Description

PBYTE shifts bytes either left or right in the PDP-9, 18-bit, word. The byte location, IB, of the input word, IW, is shift to byte location, IOB, of the output word, IOW. The bit manipulation routines are used to perform the necessary shifts. If IB is greater than IOB, a shift left is performed; if IB is less than IOB, a shift right is performed. The amount shifted, in bytes, is determined by the difference between IB and IOB.

(3) Inputs

(a) Calling Sequence - CALL PBYTE (IW, IB, IOW, IOB)

Arguments:

IW = Input word whose byte is to be shifted

IB = Byte location in input word of byte to be shifted.
1 is most significant byte, 2 is middle byte, and 3 is least significant byte.

IOW= Output word into which the shifted byte from IW is to be placed.

IOB= Byte location output word of where shifted byte from IW is to be placed in IOW. 1 is most significant byte, 2 is middle byte, and 3 is least significant byte.

NOTE: IW and IOW may be the same word in the calling program.

(4) Outputs

(a) Memory -

A PDP-9, 18 bit, word with byte shifted.

(5) Referenced Subprograms

Bit Routines: LA
 LO
 LS
 RS

k IMESG (Messages)

(1) Abstract

IMESG is the routine that handles the input tape messages to determine if additional processing is desired.

(2) Program Description

IMESG queries the user to determine if there are any further input tapes required for the present run. If there are, it issues the proper tape mount message and insures that the proper tape is mounted for processing. IMESG also switches the magnetic tape drives unit numbers which are used for the input so that continual processing may take place. IMESG also sets the done flag if no further inputs are required.

(3) Inputs

(a) Calling Sequence - CALL IMESG

Arguments: None

(b) Keyboard entries -

Y
N
D

(4) Outputs

(a) Teletype -

"ARE THERE MORE INPUT TAPES FOR THIS RUN IN
ADDITION TO THAT CURRENTLY MOUNTED ON DRIVE
_____ ENTER Y OR N."

"MOUNT NEXT INPUT TAPE ON DRIVE _____"

"ENTER D IF PROPER INPUT TAPE IS MOUNTED ON
DRIVE _____"

(b) Memory -

Done flag, located in common J1, is either set or not set depending on response to the first query.

(5) Referenced Subprogram

None

1. NPAGE (New Page)

(1) Abstract

NPAGE writes the heading information for the Selection Code Table -- Input Summary on the line printer. Information printed is the month, day, year, page number, output tape number and file number, revision number. Also printed is the heading for the table for the selection code, channel number, density and delete flag. The title is also printed at the top of the page.

(2) Program Description

NPAGE is a series of write statements to the on-line printer. The same information appears on the top of each page that is needed for the Selection Code Table - Input Summary.

(3) Inputs

(a) Calling Sequence - CALL NPAGE

Arguments: None

(4) Outputs

(a) On-line printer -

Program generates heading information on the on-line printer
"RPS SELECTION CODE TABLE - INPUT SUMMARY --/--/-- PAGE --"
"TAPE NUMBER ----"
"FILE NUMBER --"
"REVISION --"
"SELECTION CODE CHANNEL DENSITY DELETE FLAG"

(5) Referenced Subprograms

None

m. PRNSTT (Print Run Statistics)

(1) Abstract

FRNSTT prints the statistics for a given run after execution of that run has been completed.

(2) Program Description

FRNSTT prints the statistics for a given run on the on-line printed that have been accumulated for the run. These include: data; tape number; file number; elapsed time for the run in minutes and seconds; the number of scan lines input; total number of coordinates; number of designator points processed; channel totals processed for each channel.

(3) Inputs

(a) Calling Sequence - CALL FRNSTT

Arguments: None

(4) Outputs

(a) On-line Printer --

"RPS RUN STATISTICS OF mm/dd/yy"

"TAPE NUMBER -----"

"FILE NUMBER -----"

"RUN STATISTICS"

"ELAPSED TIME = -----MINUTES AND -----SECONDS"

"-----SCAN LINES INPUT CONTAINING-----X COORDINATES"

"X DESIGNATOR OF POINTS PROCESSED WERE"

CODE	COUNT
.
.
.

"CHANNEL TOTALS: CHANNEL 0 =-----, CHANNEL
1 = ----.

"CHANNEL 2 = -----, CHANNEL 3 = ---- -----".

(5) Referenced Subprograms

None

n. NSET (Set Buffer)

(1) Abstract

NSET resets the pointers to the output buffer. It also clears the buffer and presets the buffer to the high ignore value.

(2) Program Description

NSET resets the next entry counter to the beginning of the output buffer IOBUF. The LSTBYT and LSTWRD parameters are initialized to the original starting state. At the same time the output buffer is preset to the high order ignore value. This is contained in four consecutive PDP-9 words which gives us three consecutive Graphic Plotter words (24 bits each). These words have a high order ignore value, less than the EOL, such that when encountered, the plotter will go over the maximum scan position and thus not plot. This value is preset as a line center coordinate to insure a turning off of any areal data.

(3) Inputs

(a) Calling Sequence - CALL NSET

Arguments: None

(4) Outputs

(a) Memory -

A present buffer to be used as the output buffer storage for Graphic Plotter formatted data.

(5) Reference Subprograms

None

o. SETA (Set Area)

(1) Abstract

SETA is used to set the area stop coordinate when a minimum area designator is obtained from the input data.

(2) Program Description

SETA operates on the coordinate in different manners depending upon whether the coordinate is line center, area start or minimum area data. For line center data, no processing is done. The minimum flag is set to zero and control is returned to the main program. For area start, the line center indicator is removed from the coordinate control word to signify area start. The minimum flag is set to zero and again control is returned to the main program. For minimum area, the minimum area designator is masked out of the coordinate control word to simulate an area start. A line center coordinate is added to end minimum area, and the line center indicator is removed from the coordinate control word of the first word to signify area start. That is, minimum area has been converted to an area start and an area stop. Minimum area flag is set equal to one.

(3) Inputs

(a) Calling Sequence - CALL SETA

Arguments: None

(4) Outputs

(a) Memory -

An area start and area stop coordinates from minimum area settings when applicable.

(5) Referenced Subprograms

Bit Routines: LEOR
 LA

p. SETM (Set Masks)

(1) Abstract

SETM is an assembler language routine that sets the masks used by the RPS system to obtain the desired byte of a given data word.

(2) Program Description

SETM sets the labelled common MASK which contains MASK1 and MASK2. Into these locations it sequentially places the octal constants 770000, 007700, 000077, 007777, 770077, 777700. These then are used as masks to extract desired information, a byte at a time, from the data words used by the RPE system.

(3) Inputs

(a) Calling Sequence - CALL SETM (MASK)

Arguments:

MASK = the first word (starting address) of the 6 word field for which the masks will be formed.

(4) Outputs

Memory -

Preset masks.

(c) Referenced Subprograms

None

q. SETC (Set Constants)

(1) Abstract

SETC is an assembler language routine that sets the masks necessary for the output buffer area before the tape write.

(2) Program Description

SETC places the coordinate corresponding to the high order "ignore" value for the raster plotter. This value is one less than the EOL character. The masks are placed in the array INSET, used by RPSS to preset the output buffer area. The mask is four words long of the form 043700, 000437, 000001, 370000.

The masks are four, 18-bit words which form three consecutive and identical 24-bit words that is 04370000. This is the desired mask for the raster plotter.

(3) Inputs

(a) Calling Sequence - CALL SETC (INSET (1))

Arguments:

INSET = first word (starting address) of the 4 word field in which the mask will be formed.

(4) Outputs

Memory -

Preset mask.

(5) Referenced Subprograms

None

r. BITS (Bit Routines)

(1) Abstract

BITS is an assembler language subroutine with five entries that allow the assembly language logic and shifting functions to be Fortran Callable. The logic functions incorporated in the routine are:

Logical AND (LA)

Logical OR (LO)

Logical Exclusive OR (LEOR)

Left Shift (LS)

Right Shift (RS)

(2) Program Description

These functions are used by the RPS system routines that require the manipulation of bits on an integer word. For the logical functions, the routines perform the necessary operation on the first two words and place the result in the third word of the calling sequence. For the shift functions, the first word is shifted by the amount indicated in the second word and stored in the third. Operations are on a bit basis.

(3) Inputs

(a) Calling Sequence -

Logical AND: CALL LA (IA, IB, IC)

Arguments:

IA = Integer word to be AND'ed with IB

IB = Integer word to be AND'ed with IA

IC = Integer word containing result

i. e., $IA \text{ .AND. } IB \rightarrow IC$

Logical OR: CALL LO (IA, IB, IC)

Arguments:

IA = Integer word to be ORed with IB

IB = Integer word to be ORed with IA

IC = Integer word containing result

i. e., $IA \text{ .OR. } IB \rightarrow IC$

Logical Exclusive OR CALL LEOR (IA, IB, IC)

Arguments:

IA = Integer word to be EORed with IB

IB = Integer word to be EORed with IA

IC = Integer word containing result

i. e., $IA \text{ .EOR. } IB \rightarrow IC$

Left Shift: CALL LS (IA, IB, IC)

Arguments:

IA = Integer word to be shifted left.

IB = Integer word containing the number of bits to be shifted.

IC = Integer word containing result

i. e. Shift IA left IB bits and store in IC

Right Shift: CALL RS (IA, IB, IC)

Arguments:

IA = Integer word to be shifted right

IB = Integer word containing the number of bits to be shifted

IC = Integer word containing result

i. e. Shift IA right IB bits and store in IC

(4) Outputs

Memory word that is result of logical operation.

(5) Referenced Subprogram

None

s. MGFRW (Magnetic Tape Read/Write)

(1) Abstract

MGFRW is an input/output Fortran callable, assembly language utility program used to read and write blocks of data to and from standard magnetic tape.

(2) Program Description

Subprogram MGFRW is utilized by the RPS system to transfer data between core memory and standard magnetic tape. The tape handling functions performed are: Read Tape, Write Tape, Rewind Tape, and Write End-of-File (WEOF). A calling program specifies which of these functions is to be implemented by a numeric code in the calling sequence. Upon entry into MGFRW, this code and the remaining parameters needed to initiate the indicated tape operation are extracted from the calling sequence. The required instruction sequence is executed and the

program enters a "wait" loop until the I/O operation is completed. At completion of a read/write operation, MGFRW retrieves and interrogates the bit settings of the I/O status register. A unique code is stored in a predetermined location in the calling sequence to indicate the success or failure of the requested read/write operation. Control is then returned to the calling program. The calling program may proceed normally, or initiate an exception procedure, or terminate, whichever is applicable.

(3) Inputs

- (a) Calling Sequence - Call MGFRW (ARG1, ARG2, ARG3, ARG4, ARG5) all arguments are integer words and:

ARG1 = machine tape number

ARG2 = number of words to be transferred to or from memory.

ARG3 = symbolic name of memory buffer into or from which the specified number of words are to be transferred.

ARG4 = function code (right adjusted); where:

0 = Read Tape 1 = Write Tape

2 = WEOF 3 = Rewind

ARG5 = completion status code word which is initially zero and is set by MGFRW as follows:

1 = EOT (End-of-Tape) 2 = EOF (End-of-File)

3 = No Error 4 = Parity Error

5 = No MTF Flag 6 = Word Length Error

7 = Error Flag

NOTE: If ARG2 (number of words) is not equal to the physical size of the tape record read, the word length error (7) code will be stored in ARG5 of the calling sequence. If ARG2 is less than the record size, N words will be read into the buffer designated by ARG3 and the tape advanced to the end-of-record gap. If ARG2 specifies a word count greater than or equal to the record size, only

N words (record length) will be read into memory.

(b) Memory -

Data blocks of size specified by ARG2.

(4) Outputs

(a) Status Register -

Contents of status register stored in ARG5 of calling sequence. The numeric values and their meaning are:

Status Value	Meaning
1	Physical end-of-tape
2	End-of-file detected
3	Normal Termination
4	Parity error
5	Improper termination
6	Wrong word length
7	Error flag

All status values are fixed point binary numbers.

(b) Memory -

Data blocks of size specified by ARG2.

(5) Referenced Subprograms

None

2. OBJECT-TIME SYSTEM AND SCIENCE LIBRARY ROUTINES

a. Object-Time System Routines

The following is a list of PDP-9 FORTRAN IV Object-Time System macro-subroutines that are loaded at execution time. The compiled FORTRAN coded conversion programs which contain I/O statements include output calls in the form of globals to various system I/O handler routines. When the compiled object program is loaded, the compiler defined global symbols are utilized to search the FORTRAN library for the needed I/O handler routines. These are loaded into memory and the proper program control linkage established by the Link Loader. Detailed description for these programs are provided in the following DEC PDP-9 manual: "Advanced Software System Programmer's Reference Manual"; order number DEC-9A-KFZA-D.

<u>Program Name</u>	<u>Function</u>	<u>Ref. Manual</u>
AUXIO	To process the auxiliary I/O statements; BACK SPACE, REWIND, in FORTRAN IV program and subprograms.	P. 11-8
BCDIO	To process formatted READ/ WRITE statements in FORTRAN IV programs and subprograms.	P. 11-3
FIOPS	FIOPS provides the necessary call to IOPS (Input-Output Processing System) required by all FORTRAN I/O statements.	P. 11-10
GOTO (.GO)	To compute the index of a computed GO TO.	P. 11-14
OTSER (.ER)	To announce an error on the teletype.	P. 11-18
PAUSE (.PA)	To process the PAUSE statement.	P. 11-16
SPMSG (.SP)	To print the octal coded number with STOP and PAUSE.	P. 11-17
.SS	To calculate the array element address.	P. 11-12

<u>Program Name</u>	<u>Function</u>	<u>Ref. Manual</u>
STOP (.ST)	To process the STCP statement and return control to the monitor.	P. 11-15
TIME	To provide the ability to record elapsed time in minutes and seconds.	P. 11-21

b. Science Library Routines

The following PDP-9 Science Library routines are utilized by the RPSS programs. These programs are described in the DEC PDP-9 manual referenced in Section II, Subsection 2a above:

<u>Routine Name</u>	<u>External Cells</u>	<u>Ref. Manual</u>
.CB (Short Get Argument)	None	P.III-9
.DA (General Get Argument)	None	P.III-8
INTEGE (Integer Arithmetic)	REAL	P.III-8
REAL (Real Arithmetic)	None	P.III-8

SECTION III

ASSEMBLY LANGUAGE SUBPROGRAM FLOW DIAGRAMS

Preceding page blank



FIGURE IV - 6. FLOW DIAGRAM OF SETM



FIGURE IV - 7. FLOW DIAGRAM OF SETC

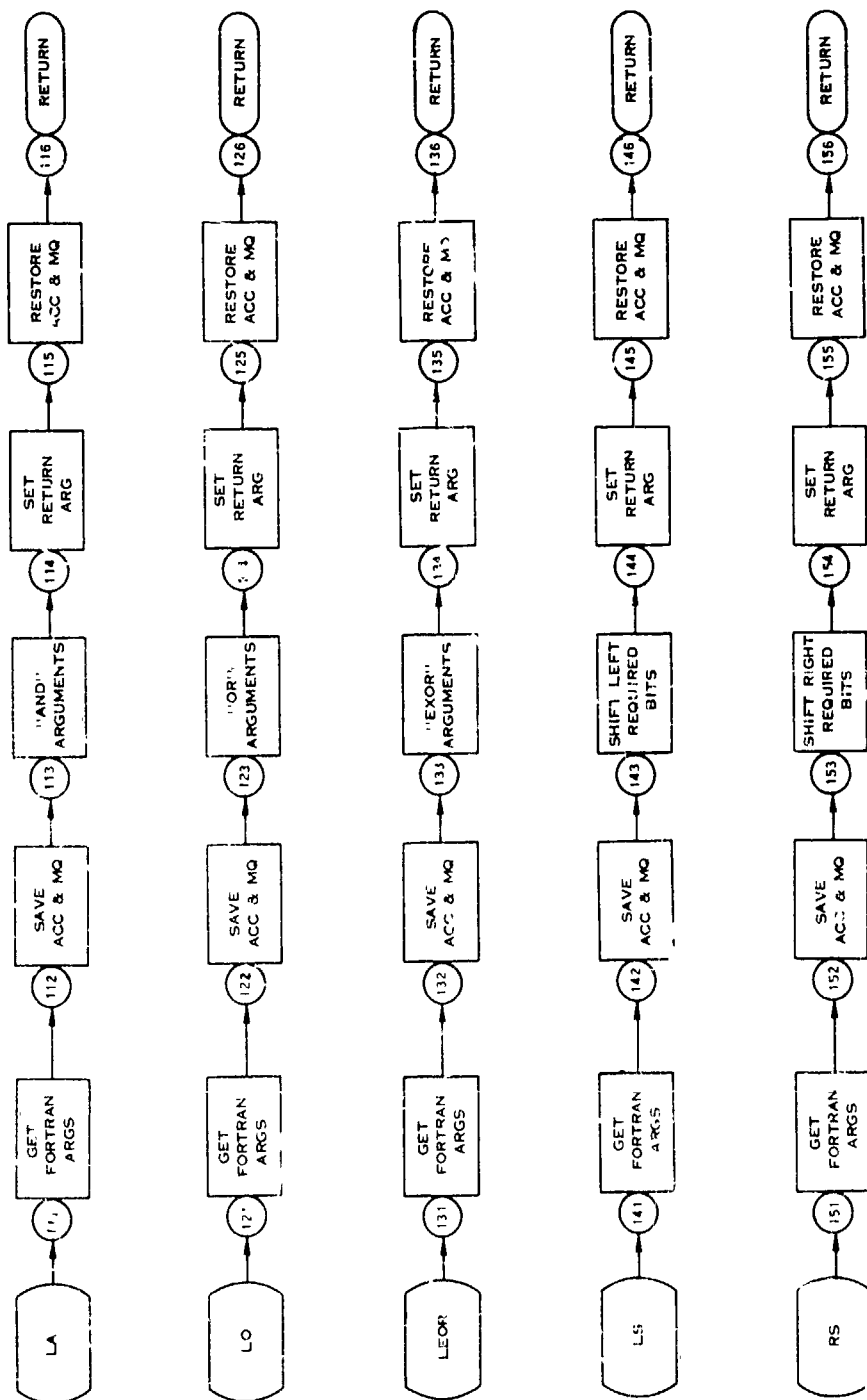


FIGURE IV - 8 FLOW DIAGRAM OF BITS

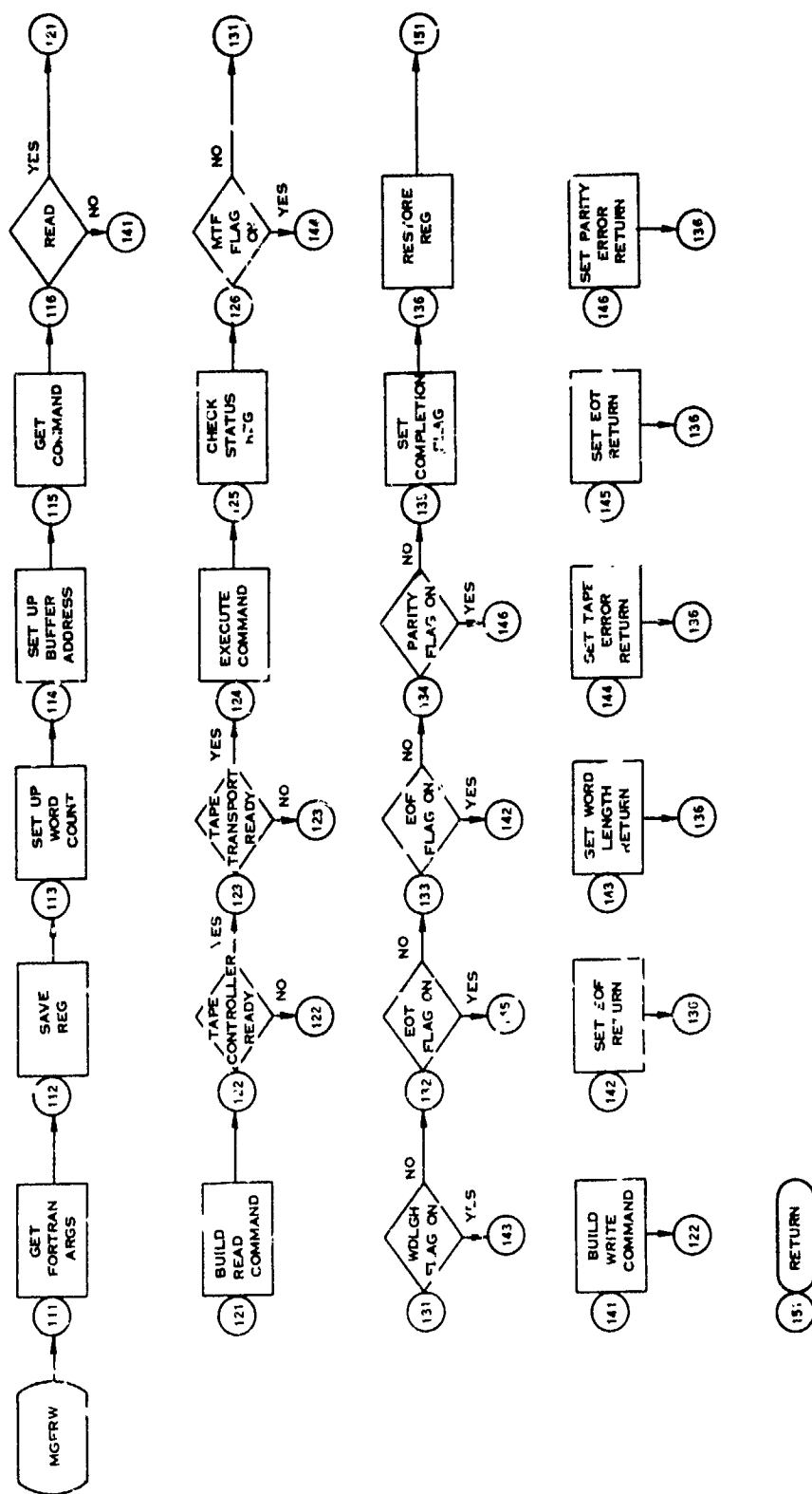


FIGURE IV - 9. FLOW DIAGRAM OF MGRW

SECTION 'V
FORTRAN SUBPROGRAM AUTOFLOW DIAGRAMS

AUTOFLOW CHART SET

07/22/71

Preceding page blank

07/22/71

INPUT LISTING

AUTOFLOW CHART SFT -

PPSS

FORTRAN MODULE (LIST,NAMGO,AUTOFLOW)

CONTENTS

```

1  DIMENSION ISC(1000,2),A(9)
2  DIMENSION MASK1(3),MASK2(3),MASKS(6)
3  DIMENSION ITBUF(740)
4  DIMENSION INSET(4),IOBUF(1364)
5  DIMENSION ICNT(25),ICHI(4)
6  COMMON /A1/ ISC,ITMAX,ITCUR,MINSC,MAXSC
7  COMMON /R1/ INSC,INCH,INDCN,INDEL
8  COMMON /C1/ IAEF,IAFCH,IAFIDN
9  COMMON /D1/ A,IDEN,ISCS,ISCE,INDEN
10 COMMON /E1/ TN,TA,TE,TX,TY,TD,TT,TS,TR,TW
11 COMMON /F1/ ITTY,IPY,IMP,IPKTR
12 COMMON /G1/ IOR1,IOR2,IOR3,IDEC
13 COMMON /H1/ INSC,ICCH,INDCN,INDEL
14 COMMON /I1/ INORTP,INAY,LYEAR,IPAGE,IFILE,IPFV
15 COMMON /J1/ INGT1,IMG12,INDNE
16 COMMON /MASK/ MASK1,MASK2
17 COMMON /GNR1/ ITYPE,ICORQ,ICONT,ICON2,NX,IX,IFT,ISTAT
18 COMMON /TRUF/ ITBUF,NYS,NXS
19 COMMON /AFLC/ IMINF,IMNAR
20 COMMON /IOUT/ IOUT1,IOUT2,IATST
21 COMMON /K1/ ICNT,ICH
22 COMMON /L1/ IMIN,ISFC,IOFF
23 COMMON /M1/ INSET,LSTHRD,LSTRT,NENT,IUHUF
24 EQUIVALENCE (MASK1(1),MASKS(1))
25 EQUIVALENCE (MASK11,MASK1(1)),(MASK12,MASK1(2)),(MASK13,MASK1(3))
26 EQUIVALENCE (MASK21,MASK2(1)),(MASK22,MASK2(2)),(MASK23,MASK2(3))

```

```

27 C
28 C INITIALIZATION
29 C
30 ISMCH = 12
31 IWRITE = 0
32 IZIV = 8
33 IOT = 4
34 IOT = 3
35 IAD = 3
36 IORIR = 1
37 IRI = 5
38 IOR2 = 6
39 IOR3 = 7
40 IORC = 3
41 IRIIC = 1
42 NX = 0
43 IV = 0
44 IFT = 0
45 ISTAT = 0
46 NVS = 0
47 NXS = 0
48 IMIN = 0
49 ISFC = 0
50 IMIN = 0
51 ISFC = 0
52 IREF = 0
53 ITMAX = 1000
54 ITCUP = 1

```


CARD NO	INPUT LISTING	CONTENTS	RDSC
07/22/71			
	0000		0000
55	MINSC = 1		
56	MAXSC = 90000		
57	ITVR = 4		
58	NRWFUS = 756		
59	IMGT1 = 1000		
60	IMGT2 = 1000		
61	IRPV = 0		
62	CALL SETM(MASKS(1))		
63	CALL SETC(INSET(1))		
64			
65	C SET UP THE RINOV CONSTANTS		
66			
67	CALL PRVTE(INSET(1),1,1000,2)		
68	TEMP = MASK11		
69	CALL RS(TEMP,1,TEMP)		
70	CALL LS(TEMP,1,TEMP)		
71	TEMP = 1		
72			
73	C OUTPUT SCRATCH MOUNT MESSAGE		
74			
75	1 WRITE (ITVR,1000) TEMP		
76	1000 FORMAT (2H MOUNT NEW SCRATCH ON DRIVE 1,1,10H ENTER 0 41EN,		
77	15H DONE)		
78	READ (ITVR,2000) A(1)		
79	2000 FORMAT (A1)		
80			
81	C CHECK		
82			

```

93      IF (A(1) .NE. 0) GO TO 1
94      C
95      C      DATE MESSAGE AND SAVE IT
96      C
97      WRITE (ITV,1001)
98
99      1001 FORMAT (40H ENTER MONTH, DAY, YEAR AS 6 NUMERICS ... M4100V)
100      READ (ITV,2001) MONTH, IDAY, IYEAR
101      2001 FORMAT (3I2)
102      C
103      C      TAPE ID MESSAGE AND SAVE IT
104      C
105      500 WRITE (ITV,1002)
106      1002 FORMAT (20H ENTER 5 DIGIT TAPE ID NUMBER)
107      READ (ITV,2002) IDT
108      2002 FORMAT (I5)
109      C
110      C      INPUT COUNT MESSAGE
111      C
112      400 WRITE (ITV,1003) ID02
113      1003 FORMAT (30H WRITE 1ST DECI OF INPUT TAPE ON DEIVE ,11)
114      400 WRITE (ITV,1101)
115      1101 FORMAT (40H ENTER 1 DIGIT MINIMUM AREA IN PROPER RESOLUTION)
116      READ (ITV,2100) I4NAR
117      2100 FORMAT (I6)
118      C
119      C      MODE OF OPERATION MESSAGE : 1 = TAPE , 2 = MANUAL
120      C
121      DO 700 J = 1,25

```

07/22/71	INPUT LISTING	AUTOFLOW CHART SFT -	RPSS
CARD NO	****	CONTENTS	****
111	700 ICNT(J) = 0		
112	00 701 J = 1.4		
113	701 ICH(J) = 0		
114	50 WRITE (ITTY,1100)		
115	1100 FORMAT (50H DO YOU WISH TO SUPPLY SELECTION CURVES EXTERNALLY / 134		
116	1 ENTER Y OR N)		
117	READ (ITTY,2000) A(1)		
118	IF (A(1) .EQ. TV) GO TO 21		
119	IF (A(1) .NE. TV) GO TO 50		
120	MODE = 1		
121	GO TO 200		
122	21 CALL SCODE		
123	MODE = 2		
124	200 CONTINUE		
125	C		
126	C START PROCESSING OF DATA		
127	C		
128	CALL IMESG		
129	CALL NSET		
130	ICURY = -1		
131	IFT = 0		
132	IMINF = 0		
133	CALL LS(TAEDEN,3,1WRK2)		
134	CALL _D(1WRK2,IAECH,1WRK2)		
135	CALL LEOR(1WRK2,4,1WRK2)		
136	CALL LS(1WRK2,6,IAECOM)		
137	ROUT1 = 0		
138	ROUT2 = 0		

```

139          TATST = 0
140          C
141          CALL TIME (IMIN,ISEC,IOFF)
142          C
143          300 IF (IMINF .NF. 1) CALL GN.
144          IF (ITYPE.GT. 2) GO TO 301
145          IF (IDONEF .EQ. 1) GO TO 300
146          CALL TMSG
147          GO TO 300
148          C
149          C NOT FOR OR ENT - CHECK IF Y OF Y
150          C IF IET = 0 THEN FIRST Y
151          C
152          301 IF (ITYPE .EQ. 3) GO TO 302
153          IF (ICORD .EQ. ICOPY) GO TO 300
154          IF (IFT .NF. 0) GO TO 300
155          IET = 1
156          GO TO 370
157          C
158          C PROCESS THE Y
159          C
160          302 IWRK1 = 0
161          IWRK2 = 0
162          C
163          C CHECK MONF: I=TAPE ; 2=SC TABLE
164          C
165          IF (MONF .EQ. 1) GO TO 360
166          C

```

CARD NO	****	CONTENTS	****
167	C	SC MODE - GET THE SELECTION CODE	
168	C		
169		CALL LA(ICOM2,MASK23,IMRK1)	
170		CALL RS(IMRK1,4,IMRK1)	
171	C		
172	C	IS IT IN SC TABLE	
173	C		
174		CALL SRCH(IMRK1,INDEX)	
175		IF (INDEX) 306,303,303	
176	C		
177	C	NOT IN TABLE,CHECK ALL ELSE	
178	C		
179		306 IF (IAFF .NE. 1) GO TO 300	
180	C		
181	C	ALL ELSE ON - SET IT UP	
182	C		
183		IMUT1 = IACOM	
184		304 IMUT2 = ICORP	
185		CALL LA(ICOM2,MASK13,IATST)	
186		ICNT(IATST) = ICNT(IATST) + 1	
187		CALL LA(IMUT1,102,IMKCH)	
188		CALL RS(IMKCH,4,IMKCH)	
189		ICN(IMKCH+1) = ICN(IMKCH+1) + 1	
190		GO TO 305	
191	C		
192	C	IN TABLE - CHECK DELETE FLAG	
193	C		
194		303 IMRK4 = ISC(INDEX,2)	

```

195      CALL LA(I=PK4,MASK13,IMRK3)
196      IF (IMRK3 .EQ. 1) GO TO 300
197      C
198      C   DELETE OFF - SET IT UP
199      C
200      CALL LA(IMRK4,MASK12,OUT1)
201      GO TO 304
202      C
203      C   TAPE MODE
204      C
205      300 CALL LA(ICUN2,MASK12,OUT1)
206      GO TO 304
207      C
208      C   MAIN STORAGE LOOP
209      C
210      305 IF (IATST .NE. 0) CALL SFTA
211      ISWCH = 313
212      IF (INENT .EQ. 567) GO TO 307
213      NENT = NENT + 1
214      C
215      C   PUT IN FIRST WORD
216      C
217      315 LSTAYT = LSTAYT + 1
218      IF (LSTAYT .NE. 4) GO TO 308
219      LSTWRD = LSTWRD + 1
220      LSTAYT = 1
221      308 CALL PBYT(I=OUT1,2,INRUFI(LSTWRD),LSTAYT)
222      C

```

07/22/71	INPUT LISTING	AUTOFLOW CHART SET -	PDSS
CARD NO	****	CONTENTS	****
223	C	NDW 3 BYTES OF SECOND	
224	C		
225		(LSTBYT .NE. 3) GO TO 309	
226		LSTWRD = LSTWRD + 1	
227		IOBUF(LSTWRD) = IOU2	
228		GO TO 310	
229	C		
230	C	NOT EVEN - ON IT 9Y BYTES	
231	C		
232		309 DO 311 KK = 1,3	
233		LSTBYT = LSTBYT + 1	
234		IF (LSTBYT .NE. 4) GO TO 312	
235		LSTWRD = LSTWRD + 1	
236		LSTBYT = 1	
237		312 CALL PRYTC(IOUT2, KK, IOBUF(LSTWRD), LSTBYT)	
238		311 CONTINUE	
239	C		
240		310 CONT. UF	
241		IF (ISWCH - 116) 313, 314, 317	
242	C		
243		313 IF (ITYPE .EQ. 3) .NR. (ITYPE .EQ. 4) GO TO 300	
244		320 CALL NSFT	
245		370 ICURY = ICORD	
246		GO TO 300	
247	C		
248	C	BUFFER FULL	
249	C		
250		307 IOU15 = IOU1	

```

251      IOUT2S = IOUT2
252      IOUT2 = IONE
253
254      IOUT1 = IOUNC
255      ISWCH = 314
256      CALL PRYTF(IOUNC,2,IOUTF(1),1)
257      GO 319 KK = 243
258
259      319 CALL PRYTF(ICIRV,KK-1,IOUTF(1),KK)
260      CALL PRYTF(ICOPY,3,IOUTF(2),1)
261      GO TO 315
262
263      C
264
265      380 IOUT2 = IENL
266      GO TO 318
267
268      314 ISTAT = 0
269      CALL MFRWFOR(I,MWRDS,IOUTF(1),1,ISTAT)
270      IWRITE = IWRITE + 1
271      IF (ISTAT .GT. 4) WRITE (R,222) IWRITE
272      222 FORMAT (23H MAX WRITE AT IWRITE = ,I5)
273      IF (ISTAT .EQ. 3 .OR. ISTAT .EQ. 4) GO TO 3
274      IF (ISTAT .NE. 1) GO TO 316
275
276      C
277      END OF TAP
278
279      C
280
281      ISWCH = 317
282      GO TO 309
283
284      317 CONTINUE
285      IOFF = 1
286      ITIME = ITIME + IMIN
287      IISFC = IISFC + ISFC
288

```


07/22/71	INPUT LISTING	AUTOFLOW CHART SFT -	RPSS
CARD NO	****	CONTENTS	****
279	IF (IISFC .LT. 60) GO TO 600		
280	IIMIN = IIMIN + 1		
281	IISFC = IISFC - 60		
282	600 CONTINUE		
283	IOFF = 0		
284	REWIND FOR1		
285	WRITE (ITTY,1050) IDRI		
286	1050 FORMAT (40H AT PAUSE 2, MOUNT NEW SCRATCH ON DRIVE ,11,14H AND ON		
287	INTINUE)		
288	C		
289	PAUSE 2		
290	CALL TIME (IMIN,ISEC,IOFF)		
291	C		
292	316 IF (ITYPC .LT. 2) GO TO 301		
293	IF (ITYPC .EQ. 4) GO TO 320		
294	CALL NSFT		
295	IOUT1 = IOUT15		
296	IOUT2 = IOUT25		
297	GO TO 305		
298	C		
299	390 IOUT2 = IECM		
300	WRITE (B,5000) IWRITE		
301	5000 FORMAT (16H TOTAL WRITES = ,I5,4H + 1)		
302	IWRITE = 0		
303	GO TO 318		
304	C		
305	391 GO TO 400		
316	C		

307	C	OUTPUT TAPE ERROR
308	C	
309		350 STOP 12
310	C	
311	C	PRINT OUTPUT STATISTICS
312	C	
313		400 IOFF = 1
314		IMIN = IMIN + IIMIN
315		ISEC = ISEC + IISEC
316		IF (IISEC -LT. 60) GO TO 7000
317		IMIN = IMIN + 1
318		ISEC = ISEC - 60
319		7000 CONTINUE
320		YIMIN = 0
321		IISEC = 0
322		CALL PRNSIT
323		IOFF = 0
324		IFLG2 = 0
325	C	
326	C	RUN TERMINATION MESSAGE
327	C	
328		WRITE (ITTY,1200)
329		1200 FORMAT (42H DO YOU WISH TO TERMINATE RUN (Y OR N))
330		READ (ITTY,2000) A111
331		IF (A111) .EQ. 'Y') GO TO 401
332	C	
333	C	CONTINUE RUN - MULTIPLE MESSAGE
334	C	

CARD NO	INPUT LISTING	AUTOFLOW CHART SET -	PAGE
	*****	CONTENTS	*****
335	WRITE (ITTY,1201)		
336	1201 FORMAT (47H DO YOU WISH TO STACK OUTPUT FILES (Y OR N))		
337	READ (ITTY,2000) A11)		
338	IF (A11) .EQ. TV) GO TO 402		
339	C		
340	C NEW OUTPUT TAPE REQUIRED		
341	C		
342	IFLG2 = 1		
343	401 REWIND IMGT2		
344	CALL MGRM(IOR1,NRMRDS,ORJF(1),2,(OSTAT)		
345	REWIND IDRI		
346	C		
347	IMGT1 = IOR2		
348	IMGT2 = IOR3		
349	C		
350	C TAPE REMOVAL MESSAGE		
351	C		
352	WRITE (ITTY,1202) TOR1,IMONTH,IDAY,IYEAR,IODT,IRFEL		
353	1202 FORMAT (28H REMOVE OUTPUT TAPE (W DRIVE,1X,11/ 15H LABEL IT DATE,		
354	14X,12,1H/,12,1H/,12,1H) TAPE NUMBER ,14,14H REEL NUMBER ,12)		
355	C		
356	C CHECK IF STACKED OUTPUT REQUIRED		
357	C		
358	IF (IFLG2 .NE. 1) GO TO 999		
359	GO TO 403		
360	C		
361	C STACK OUTPUT		
362	C		

```

363      402 CALL "GFRW(IORT,MRWRS, IORUF(1),2,I(STAT)
364          REMIND INGT?
365      C
366      C CHECK MODE
367      C
368      403 CONTINUE
369          IF (MODE .EQ. 1) GO TO 404
370      C
371      C SELECTION CODE MODE - - SAME SC MESSAGE
372      C
373          WRITE (TTY,1203)
374      1203 FORMAT (61H DO YOU WISH TO USE THE SAME SELECTION CODES AS PREVIOUS
375          IS RUN / 11H (Y OR N))
376          READ (TTY,2000) A(1)
377          IF (A(1) .NE. 'Y') GO TO 404
378      C
379      C CHECK FOR STACKED OUTPUT
380      C
381          IF (IFLG2 .EQ. 1) GO TO 405
382      C
383      C CHECK FOR FOT - - STACK OUTPUT
384      C
385          IF (I(STAT) .NE. 1) GO TO 406
386      C
387      C OUTPUT TAPE FINISHED - CANNOT STACK - NEED NEW OUTPUT TAPE
388      C
389          405 WRITE (TTY,1000) (NO)
390          READ (TTY,2000) A(1)

```

CARD NO	INPUT LISTING	AUTOFLOW CHART SFT -	POSS
	****	CONTENTS	****
391	IF(A11) .NE. TD) GO TO 405		
392	WRITE (ITTY,1001)		
393	READ (ITTY,2001) MONTH,JDAY,IYEAR		
394	WRITE (ITTY,1002)		
395	READ (ITTY,2002) INT		
396	404 WRITE (ITTY,1003)IDR2		
397	GO TO 200		
398	404 CONTINUE		
399	C		
400	C COMPLETELY NEW INPUT, FORMAT - CHECK FOR STACKED OUTPUT		
401	C		
402	IF (ICLG2 .EQ. 1) GO TO 1		
403	GO TO 500		
404	C		
405	C TERMINATION PROCEDURE - - TERMINATION MESSAGE		
406	C		
407	999 WRITE (ITTY,1204)		
408	1204 FORMAT (424 END OF RUN - REMOVE ALL TAPES FROM DRIVE)///END,END **		
409	1* TERMINATION ***)		
410	C		
411	C		
412	STOP		
413	END		
414	SUBROUTINE SCORE		10
415	C		20
416	C *** PART OF THE QPS SYSTEM THAT HANDLES THE SELECTION CODE PROCESSING		30
417	C *** OF INPUT DATA. THIS SUBROUTINE ACCEPTS SELECTION CODES, IDENTITIES		40
418	C *** AND CHANNEL NUMBERS THAT WILL BE THE SELECTION CRITERIA FOR THE		50

```

419 C *** FINAL OUTPUT PRODUCT (NCE PROCESSING IN PHASE IT IS COMPLETED. 40
420 C *** THIS IS AN OPTION IN ADDITION TO THE MEANS OF PROCESSING THE DATA 70
421 C *** DIRECTLY FROM THE INPUT TAPE 80
422 C 90
423 DIMENSION ISC(1000,2),A(9) 100
424 COMMON /A1/ ISC,ITMAX,ITCIP,MINSC,MAXSC 120
425 COMMON /B1/ INSC,INCH,INDEL 130
426 COMMON /C1/ TAFF,IAECH,IAFDEN 140
427 COMMON /D1/ A,INDEN,ISCS,ISCF,INDEN
428 COMMON /E1/ TN,TA,TE,FX,FY,TD,TP,TT,TS,TU,TW
429 COMMON /F1/ ITTY,IPT,INT,INP,IPRTR
430 COMMON /G1/ IND1,IND2,IND3,INFC
431 COMMON /H1/ INSC,INCH,INDEN,INDEL 190
432 COMMON /I1/ IMONTH,IDAY,IYEAR,IPAGE,IFILF,IPRV 210
433 C
434 WRITE (ITTY,1004) 260
435 1004 FORMAT (45H ENTER DEFAULT DENSITY AS NUMERIC FROM 0 TO 7)
436 C
437 C *** READ DENSITY
438 C
439 READ (ITTY,2003) INDEN
440 2003 FORMAT (I1)
441 C
442 C *** PRESET SC TABLE
443 C
444 INSC = 0
445 INCH = 0
446 INDEL = 0

```

07/22/71	INPUT LISTING	AUTOFLOW CHAPT SET -	ROSS
CARD NO	****	CONTENTS	****
447	ICH = 0		
448	DO ? J = 1,ITMAX		
449	2 CALL INTMDL(J)		
450	C		
451	C *** CLEAR ALL ELSE FUNCTION		
452	C		
453	IREV = 0		
454	IAEF = 0		
455	IAECH = 5		
456	IAEDEN = IDDEN		
457	C		500
458	9 WRITE (ITTY,1009)		510
459	1009 FORMAT (51H ENTER CHANNEL NUMBER FROM 1 TO 4 OR ENTER 0 TO END/ 52		
460	1H SELECTION CODE INPUT EACH TIME CH NR = 15 TYPED)		
461	11 WRITE (ITTY,1011)		
462	1011 FORMAT (8H CH NR =)		
463	READ (ITTY,2003) INCH		
464	IF (INCH .EQ. 0) GO TO 100		
465	INCH = INCH - 1		
466	IF (INCH .GE. 4) GO TO 9		
467	C		
468	C *** AUTOMATIC/MANUAL MESSAGE		
469	C		
470	INP = ITTY		
471	WRITE (ITTY,1012)		
472	1012 FORMAT (12H ENTER CODES,2X,15H(WYZNNNNNNNNNN))?		
473	C		
474	C *** PUT THE ENTRIES IN THE TABLE		

```

475 C
476     NR = 0
477     MINS1 = -1
478     MINS2 = -2
479     MINS3 = -3
480     MINS4 = -4
481     10 CALL MKENT(NR)
482     IF (NR .GE. 0) GO TO 10
483     IF (NR .EQ. MINS1) GO TO 11
484     IF (NR .EQ. MINS2) GO TO 10
485     IF (NR .EQ. MINS3) GO TO 100
486     IF (NR .EQ. MINS4) PAUSE 10
487     GO TO 10
488 C
489 C *** INPUT COMPLETE - SORT THE TABLE
490 C
491     100 CALL SHLSRT
492 C
493 C *** PRINT THE TABLE
494 C
495     IOELT = 0
496     CALL NPAGE
497     LCNT = 0
498     ICNT = ITCUR - 1
499     DO 15 I = 1, ICNT
500 C
501 C *** OBTAIN PARAMETERS FROM SC TABLE
502 C

```


07/22/71	INPUT LISTING	AUTOFLOW CHART SFT -	PPSS
CARD NO	****	CONTENTS	****
503	CALL OUTMDL(1)		
504	WRITE (IPTR,1018) IOSC,IOCH,INMEN,IODEL		
505	1018 FORMAT (6X,15,15X,12,11X,11,12X,11)		
506	LCNT = LCNT + 1		
507	IF (LCNT .LT. 40) GO TO 15		
508	LCNT = 0		
509	IPAGE = IPAGE + 1		
510	CALL NPAGE		
511	15 CONTINUE		
512	C		
513	C *** CHECK ALL ELSE AND OUTPUT IF ON		
514	C		
515	IF (IAEF .NE. 1) GO TO 14		
516	WRITE (IPTR,1018) IAECM,IAEDEN		
517	1019 FORMAT (5X,8HALL ELSE,13X,12,11X,11)		
518	C		
519	C *** SUMMARY MESSAGE		
520	C		
521	14 WRITE (ITTY,1020)		
522	1020 FORMAT (61H SELECTION CODE SUMMARY ON PRINTER. ARE CORRECTIONS REQ		
523	UIRED / 14H ENTER Y OR N.)		
524	C		
525	C *** READ ANSWER		
526	C		
527	READ (ITTY,2000) A(1)		
528	IF (A(1) .NE. TV) GO TO 14		
529	C		
530	C *** CORRECTIONS REQUIRED		

```

531 C
532 IREV = IREV + 1
533 GO TO 9
534 16 CONTINUE
535 IF (A(1) .NE. TN) GO TO 14
536 C
537 C *** NU CORRECTIONS - - - RETURN
538 C
539 RETURN
540 END
541 SUBROUTINE MRENT(IFLG)
542 C
543 C *** READS THE INPUT SELECTION CODE AND ENTERS IT INTO THE
544 C *** SELECTION CODE TABLE
545 C
546 DIMENSION ISC(100,2), A(9)
547 COMMON /A1/ ISC,ITMAX,ITCUR,PINSC,MAXSC
548 COMMON /A1/ INSC,INCH,INDEN,INDEL
549 COMMON /C1/ IAEF,IAECH,IAEDEN
550 COMMON /D1/ A,IDEN,ISCS,ISCF,IDEN
551 COMMON /E1/ TN,TA,TE,TX,TY,TD,TP,TT,TS,TR,TM
552 COMMON /F1/ ITTY,IPT,IDT,INP,IPTR
553 COMMON /G1/ IDR1,IDR2,IDR3,INFC
554 COMMON /H1/ IOSC,IOCH,IODEN,INDEL
555 C
556 IFLG = 0
557 C
558 C *** READ THE ENTRY

```

CARD NO	INPUT LISTING	AUTDFLOW CHART SET -	POSS
559	****	CONTENTS	****
560	C		
561	REFA, IINP, 2004) A(1), A(2), IDFN, ISCS, ISCF		
562	2004 FORMAT (A1, A1, I1, 2(5)		
563	C		
564	C *** TEST FOR END		
565	C		
566	IF (A(1) .NE. TX) GO TO 1		
567	IFLG = -1		
568	RETURN		
569	C		
570	C *** NOT END - - - CHECK THE ENTRY		
571	C *** ERROR CHECK CODE		
572	C		
573	1 IF ((A(1) .NE. TA) .AND. (A(1) .NE. TF) .AND. (A(1) .NE. TX) .AND.		
574	1 (A(1) .NE. TN)) GO TO 4		
575	IF ((A(2) .NE. TS) .AND. (A(2) .NE. TR)) GO TO 4		
576	IF ((IDFN .LT. 0) .OR. (IDFN .GT. 8)) GO TO 4		
577	IF ((ISCS .LT. MINSC) .OR. (ISCS .GT. MAXSC)) GO TO 4		
578	IF ((TA(2) .EQ. TR) .AND. (ISCF .LT. INSC) .OR. (ISCF .GT. MAXSC)))		
579	1 GO TO 4		
580	C		
581	C *** ENTRY OK --- NOW DETERMINE THE TYPE		
582	C		
583	C		
584	IF (A(1) .NE. TA) GO TO 2		
585	C		
586	C *** ENTRY IS ALL ELSE		

```

597       IAEF = 1
598       IAECH = INCH
599       IF (IDEN .NE. 0) IADEN = IDEN
600       RETURN
601       C
602       C *** NOT ALL ELSEF - - - MUST BE AN AND OR DELFTF
603       F
604       2 INDEL = 0
605       IF (A(1) .EQ. TE) INDEL = 1
606       C
607       C *** SFT UP FOR INTWDL - PRESFT DEFAULT DENSITY AND CHANGE IF NECESSARY
608       C
609       INDEN = IDEN
610       IF (IDEN .GT. 0) INDEN = IDFN
611       C
612       C *** SFT UP DD LOOP
613       C
614       JJ = ISCS
615       KK = ISCE
616       IF (A(2) .NE. TR) KA = ISCS
617       C
618       C *** DD LOOP FURMS THE TABLE
619       C
620       DO 6 I = JJ, KK
621       INSC = I
622       CALL SCAN(I, K)
623       C
624       C *** CHECK FOR PREVIOUS ENTRY

```

07/22/71	INPUT LISTING	AUTOFLOW CHART SET -	RPSS
CARD NO	****	CONTENTS	****
615	C		
616		IF (K .GT. 0) GO TO 7	
617	C		
618	C ***	NO PREVIOUS ENTRY	
619	C		
620		K = ITCUR	
621		ITCUR = ITCUR + 1	
622	C		
623	C ***	IF PREVIOUS ENTRY, OVERWRITE, IF NOT ADD AT END OF TABLE	
624	C		
625		7 CALL INTWDL(K)	
626	C		
627	C ***	CHECK IF TABLE FULL	
628	C		
629		IF (ITCUR .GT. ITMAX) GO TO 8	
630	C		
631	C ***	TABLE NOT FULL	
632	C		
633		6 CONTINUE	
634		IF (IMP .EQ. ITTY) WRITE (ITTY,1012)	
635		1012 FORMAT (3H OK)	
636		IFLG = K	
637		RETURN	
638	C		
639	C ***	TABLE FULL - STOP ENTERING - WRITE LAST ENTRY MESSAGE	
640	C		
641		8 WRITE (ITTY,1010) A(1),A(2),IDEN,ISCS,ISCE	
642		1010 FORMAT (39H SELECTION CONF TABLE FULL - LAST ENTRY,	

07/22/71	INPUT LISTING	AUTOFLOW CHART SET -	HPSS
CARD NO	****	CONTENTS	****
671	ITAPE = IMGT2		
672	CALL PFOF(ITAPE)		
673	3 CONTINUE		
674	CALL MGRW(IMGT2,NRWD,ITBUF(1),0,ISTAT)		
675	IREAD = IREAD + 1		
676	IF (ISTAT .EQ. 1 .OR. ITBUF(1) .EQ. 6140) GO TO 4		
677	IF (ISTAT .GT. 4) GO TO 2		
678	C		
679	ITYPE = ISTAT		
680	7 IX = 14		
681	ICORD = ITBUF(10)		
682	ICON1 = ITBUF(11)		
683	IF (IT .EQ. 9) WRITE (9,1010) ITBUF(10)		
684	1010 FORMAT (23H FIRST Y COORDINATE IS ,16)		
685	ICON2 = ITBUF(12)		
686	NK = ITBUF(6)		
687	NYS = NYS + 1		
688	NXS = NXS + NK		
689	C		
690	C		
691	C		
692	IF (ITYPE .EQ. 3) ITYPE = 4		
693	IF (ITYPE .EQ. 4) RETURN		
694	C		
695	1 IF (NK .EQ. 0) GO TO 3		
696	IF (IX .GT. 640) GO TO 4		
697	5 ICORD = ITBUF(IX)		
698	ICON1 = ITBUF(IX+1)		

```

699      ICON2 = ITRBUF(1X+2)
700      IX = IX + 4
701      ITYPE = 3
702      C
703      C      TYPE IS X
704      C
705      NX = NX - 1
706      RETURN
707      C
708      4 CONTINUE
709      CALL MGRFW(1XCT2,NRWRDS,ITRBUF(1),0,ISTAT)
710      IREAD = IREAD + 1
711      IF (ISTAT .GT. 4) WRITE (9,1020) IREAD
712      IF (ISTAT .EQ. 1) .OR. ITRBUF(1) .EQ. 61440) GO TO 4
713      IX = 6
714      GO TO 5
715      C
716      2 ITYPE = 5
717      ITEST = -37450
718      IF (ISTAT .EQ. 4) ITYPE = 3
719      IF (ITRBUF(7) .EQ. ITEST) ITYPE = 4
720      IF (ITYPE .EQ. 5) WRITE (9,1020) IREAD
721      IF (ITRBUF(7) .EQ. ITEST) GO TO 7
722      1020 FORMAT (21H BAD READ AT IREAD = ,I5)
723      RETURN
724      C
725      C      END OF EDIT FUNCTION
726      C

```


07/22/71	INPUT LISTING	AUTOFLOW CHART SET -	RPSS
CARD 40	****	CONTENTS	****
727	4 ITYPE = ISTAT		
728	IF (ITATF(1) .EQ. 61440) ITYPE = 2		
729	WRITE (8,1030) IREAD		
730	1030 FORMAT (15H TOTAL READS = ,I5)		
731	RETURN		
732	END		
733	SUBROUTINE SCAN(IVAL,K)		
734			
735	C *** SCANS SELECTION CODE TABLE FOR DUPLICATE ENTRIES		
736			
737	DIMENSION ISC(1000,2)		
738	COMMON /A1/ ISC,ITMAX,ITCUR,MINSC,MAXSC		
739			
740	JJ = ITCUR - 1		
741	DO 1 I = 1, JJ		
742	IF (ISC(I,1) .NE. IVAL) GO TO 1		
743			
744	C *** DUPLICATE FOUND		
745			
746			
747	RETURN		
748	1 CONTINUE		
749			
750	C *** NO DUPLICATE - I.E. NOT IN TABLE		
751			
752	K = -1		
753	RETURN		
754	END		

```

755      SURROUTINE SHLSRT
756      C
757      C *** SORT THE SELECTION CODE TABLE IN INCREASING ORDER BY FIRST WORD
758      C
759      DIMENSION ISC(1000,2),IB(1,2)
760      COMMON /A1/ ISC,ITMAX,ITCUR,MINSC,MAXSC
761      C
762      LN = ITCUR - 1
763      1 LN = LN/2
764      IF (LN .EQ. 0) GO TO 2
765      C
766      C *** CONTINUE SORT
767      C
768      J = 1
769      K = ITCUR - 1 - LN
770      3 I = J
771      4 IN = I + LN
772      IF (ISC(I,1) .LE. ISC(IN,1)) GO TO 5
773      C
774      C *** PERFORM EXCHANGE
775      C
776      DO 6 JJ = 1,2
777      1A(1,JJ) = ISC(I,JJ)
778      ISC(I,JJ) = ISC(IN,JJ)
779      6 ISC(IN,JJ) = 1A(1,JJ)
780      I = I - LN
781      IF (I .GE. 1) GO TO 4
782      C

```

CARD NO	INPUT LISTING	AUTOFLOW CHART SET -	PPSS	CONTENTS	*****
793	C *** IN CORRECT ORDER				
784	C				
785	5 J = J + 1				
786	C				
787	C *** MORE TO SORT				
788	C				
789	IF (J - K) 3,3,1				
790	C				
791	C *** SORT FINISHED				
792	C				
793	2 CONTINUE				
794	RETURN				
795	END				
796	SUBROUTINE SRCH(ARG,K)				
797	C				
798	C *** PERFORMS A BINARY SEARCH OF THE SELECTION CODE TABLE				
799	C				
800	DIMENSION TSC(1000,2)				
801	COMMON /A1/ ISC,ITMAX,ITCUR,MINSC,MAXSC				
802	C				
803	IFIRST = 1				
804	ILAST = ITCUR - 1				
805	1 IF (ILAST - IFIRST) .LT. 0) GO TO 4				
806	LN = (ILAST + IFIRST)/2 + IFIRST				
807	IF (IARG .EQ. TSC(LN,1)) GO TO 2				
808	IF (IARG .GT. TSC(LN,1)) GO TO 3				
809	C				
810	C *** VALUE IN FIRST PART OF TABLE SECTION BEING SEARCHED				

```

011 C
012 ELAST = LN - 1
013 GO TO 1
014 3 IFST = LN + 1
015 C
016 C *** VALUE IN SECOND PART OF TABLE SECTION
017 C
018 GO TO 1
019 C
020 C *** VALUE FOUND IN TABLE
021 C
022 2 K = LN
023 RETURN
024 C
025 C *** VALUE NOT IN TABLE
026 C
027 4 K = -1
028 RETURN
029 END
030 SUBROUTINE INTWOL(K)
031 C
032 C *** SETS UP THE SELECTION CODE TABLE ENTRIES
033 C *** FIRST WORD IS THE SELECTION CODE
034 C *** SECOND WORD BYTE 1 IS ZERO
035 C *** BYTE 2 IS DENSITY AND CHANNEL NUMBER
036 C *** BYTE 3 IS DELETE FLAG
037 C
038 DIMENSION ISC(1000,2)

```

07/22/71	INPUT LISTING	AUTOFLOW CHART SET -	PASS
CARD NO	*****	CONTENTS	*****
R39	COMMON /A1/ ISC,ITMAX,ITCUP,MINSC,MAXSC		
R40	COMMON /B1/ INSC,INCH,INDEFN,INDEL		
R41	C		
R42	C *** CLEAR TABLE ENTRY		
R43	C		
R44	ISC(K,1) = 0		
R45	ISC(K,2) = 0		
R46	C		
R47	C *** SET FIRST WORD		
R48	C		
R49	ISC(K,1) = INSC		
R50	C		
R51	C *** FORM SECOND BYTE OF SECOND WORD		
R52	C		
R53	ITEMP = 0		
R54	CALL LS(INDEFN,1,ITEMP)		
R55	CALL LO(ITEMP,INCH,ITEMP)		
R56	CALL LFOR(ITEMP,4,ITEMP)		
R57	CALL LS(ITEMP,4,ITEMP)		
R58	ITEMP = ITEMP + INDEL		
R59	C		
R60	C *** SET SECOND WORD		
R61	C		
R62	ISC(K,2) = ITEMP		
R63	RETURN		
R64	END		
R65	SUBROUTINE AUTWOLIK		
R66	C		

```

867 C *** DECODE THE SELECTION CODE TABLE ENTRIES
868 C
869     DIMENSION ISC(1000,2)
870     COMMON /A/ ISC,ITMAX,ITCUR,MINSC,MAXSC
871     COMMON /H1/ INSC,IOCH,INDEN,INDFL
872 C
873 C *** OBTAIN SELECTION CODE
874 C
875     INSC = ISC(K,1)
876 C
877 C *** STRIP SECOND WORD
878 C
879     ITEMP = ISC(K,2)
880     CALL LA(ITEMP,1,INDFL)
881     CALL RS(ITEMP,4,ITEMP)
882     CALL LA(ITEMP,56,INDEN)
883     IOCH = ITEMP - INDEN
884     CALL RS(INDEN,3,INDEN)
885     IOCH = IOCH - 4
886     RETURN
887     END
888     SUBROUTINE PRYF(IW,IO,ION,IOB)
889 C
890 C *** SHIFTS BYTES FROM A GIVEN LOCATION IN ONE WORD TO A GIVEN
891 C *** LOCATION IN ANOTHER WORD
892 C ***     IW = INPUT WORD
893 C ***     IO = BYTE LOCATION OF INPUT WORD TO BE SHIFTED - 1,2 OR 3
894 C ***     OI = OUTPUT WORD

```

CARD NO	INPUT LISTING	AUTOFLOW CHART SET -	RPSS	CONTENTS	****
895	C ***			IOB = BYTE LOCATION TO BE SHIFTED INTO IN OUTPUT WORD-1,2,3	****
896				DIMENSION MASK1(3),MASK2(3),MASKS(6)	
897				COMMON /MASK/ MASK1,MASK2	
898				EQUIVALENCE (MASKS(1),MASK1(1))	
899	C				
900				CALL LA(IW,MASK1(I01),IWRK1)	
901				CALL LA(IOW,MASK2(I0B),IWRK2)	
902				IX = 6*(I0 - I0B)	
903				IF (IX) 1,2,3	
904				1 IX = -IX	
905				CALL PS(IWRK1,IX,IWRK1)	
906				2 CALL LO(IWRK1,IWRK2,IOW)	
907				RETURN	
908				3 CALL LS(IWRK1,IX,IWRK1)	
909				GO TO 2	
910				END	
911				SUBROUTINE IMESG	
912	C				
913	C ***			WRITES INPUT TAPE MESSAGES	
914	C				
915				COMMON /EI/ IN,TA,TE,TX,TY,TD,TP,TT,TS,TQ,TW	
916				COMMON /FI/ ITTY,IPT,IOT,IMP,IPTR	
917				COMMON /GI/ IDR1,IDR2,IDR3,IDEC	
918				COMMON /JI/ IMGT1,IMGT2,IDONE	
919	C				
920	C ***			MORE TAPE MESSAGE	
921	C				
922				WRITE (ITTY,I018) IMGT1	

```

923 '018 FORMAT (54H ARE THERE MORE INPUT TAPES FOR THIS RUN IN ADDITION T,
924 16H0 THAT/ 27H CURRENTLY MOUNTED ON DRIVE,1X,11,15H ENTER Y OR N,
925 C
926 C *** READ ANSWER
927 C
928 READ (ITTY,2015) ANS
929 2015 FORMAT (A1)
930 C
931 C *** SET DONE FLAG IF NO
932 C
933 IDONE = 0
934 IF (ANS .NE. TY) IDONE = 1
935 C
936 C *** CHECK IF DONE
937 C
938 IF (IDONE .EQ. 1) GO TO 1
939 C
940 C *** MOUNT NEXT TAPE MESSAGE
941 C
942 WRITE (ITTY,1019) INGT2
943 1019 FORMAT (32H MOUNT NEXT INPUT TAPE ON DRIVE .11)
944 C
945 C *** PROPER TAPE MOUNTED MESSAGE
946 C
947 1 WRITE (ITTY,1020) INGT1
948 1020 FORMAT (50H ENTER 0 IF PROPER INPUT TAPE IS MOUNTED ON DRIVE .11)
949 READ (ITTY,2015) DON
950 IF (DON .NE. TD) GO TO 1

```


07/22/71	INPUT LISTING	AUTOFLOW CHART SET -	RPSS
CARD NO	****	CONTENTS	****
951	C		
952	C ***	PROPER TAPE MOUNTED - RESET DRIVE NUMBERS AND THEN RETURN	
953	C		
954		IF (IMGT1 .EQ. IOR3) GO TO 2	
955		IMGT1 = IOR3	
956		IMGT2 = IOR2	
957		RETURN	
958		2 IMGT1 = IOR2	
959		IMGT2 = IOR3	
960		RETURN	
961		END	
962		SUBROUTINE NPAGE	
963	C		
964	C ***	OUTPUTS THE HEADING INFORMATION FOR THE SELECTION CODE LIST	
965	C		
966		COMMON /F1/ ITTY, IPT, IDT, IMP, IPRTR	
967		COMMON /I1/ IMONTH, IDAY, IYEAR, IPAGE, IFILE, IREV	
968	C		
969		WRITE (IPRTR, 1013) IMONTH, IDAY, IYEAR, IPAGE	
970		1013 FORMAT (47H1 RPS SELECTION CODE TABLE - INPUT SUMMARY ,12,1H/,	
971		112,1H/,12,10H PAGE ,12)	
972		WRITE (IPRTR, 1014) IOT	
973		1014 FORMAT (14H0 TAPE NUMBER ,15)	
974		WRITE (IPRTR, 1015) IFILE	
975		1015 FORMAT (14H FILE NUMBER ,12)	
976		WRITE (IPRTR, 1016) IREV	
977		1016 FORMAT (14H REVISION ,12)	
978		WRITE (IPRTR, 1017)	

	1017 FORMAT (53HD SELECTION CODE	CHANNEL	DENSITY	DFLET,
979	16HE FLAG)			
980	RETURN			
981	END			
982	SUBROUTINE PRNSTY			
983				
984				
985	C *** PRINT RUN STATISTICS			
986				
987	DIMENSION ICNT(25),ICNT(4),ITBUF(740)			
988	COMMON /K1/ ICNT,ICM			
989	COMMON /TBUF/ ITBUF,NYS,NXS			
990	COMMON /L1/ IMIN,ISEC,IOFF			
991	COMMON /F1/ IYTY,IPT,INT,INP,IPRTR			
992	COMMON /I1/ IMONTH,IDAY,IYEAR,IPAGE,IFILE,IRFV			
993				
994	WRITE (IPRTR,1300) IMONTH,IDAY,IYEAR			
995	1300 FORMAT (24H1 RPS RUN STATISTICS OF ,I2,IH/,I2,IH/,I2)			
996	WRITE (IPRTR,1301) INT			
997	1301 FORMAT (14H0 TAPE NUMBER ,I5)			
998	WRITE (IPRTR,1302) IFILE			
999	1302 FORMAT (14H FILE NUMBER ,I2)			
1000				
1001	C STATISTICS			
1002				
1003	WRITE (IPRTR,1303) (MIN,ISEC,NYS,NXS,(K,ICNT(K),K=1,25)			
1004	1303 FORMAT(15H RUN STATISTICS//16H ELAPSED TIME = ,I6,IH,12H)MUTFS A			
1005	IND ,I6,IH,7HSECONDS////I6,IH,27HSCAN LINES INPUT CONTAINING,IH,I6,			
1006	214H X COORDINATES////I30H X DESIGNATOR OH POINTS PROCESSED WEPF///2			

07/22/71	INPUT LISTING	AUTOFLOW CHART SET -	PPSS
CARD NO	*****	CONTENTS	*****
1007	15X,4CODE,12X,5MCOUNT,25(1/25X,14,12X,16))		
1008	WRITE (IPRTR,1304) (ICH(K),K=1,4)		
1009	1304 FORMAT (30MCCHANNEL TOTALS: CHANNEL 0 = ,16,1X,13M CHANNEL 1 = ,1		
1010	16,1X,13M CHANNEL 2 = ,16,1X,13M CHANNEL 3 = ,16)		
1011	RETURN		
1012	END		
1013	SUBROUTINE MSET		
1014	C		
1015	C *** RESETS OUTPUT BUFFER POINTERS AND CLEARS THE BUFFER TO THE HIGH		
1016	C *** IGNORE VALUE		
1017	DIMENSION INSET(4),IOBUF(1364)		
1018	COMMON /M1/ INSET,LSTWRD,LSTBYT,MENT,IOBUF		
1019	C		
1020	MENT = 1		
1021	LSTBYT = 1		
1022	LSTWRD = 2		
1023	DO 1 J = 1,756,4		
1024	DO 2 K = 1,4		
1025	KK = J + K - 1		
1026	2 IOBUF(KK) = INSET(K)		
1027	1 CONTINUE		
1028	RETURN		
1029	END		
1030	SUBROUTINE SETA		
1031	C		
1032	C ***		
1033	C		
1034	DIMENSION MASK1(3),MASK2(3)		

1035	COMMON /GHR1/ ITYPE,ICORD,ICON1,ICON2,NX,IX,IFT,ISTAT
1036	COMMON /AFLC/ IMINF,IMNR
1037	COMMON /MASK/ MASK1,MASK2
1038	COMMON /IOUT/ IOUT1,IOUT2,IATST
1039	EQUIVALENCE (MASK13,MASK1(3))
1040	
1041	IMINF = 0
1042	IF (IATST .GT. 3) GO TO 2
1043	GO TO (1,2,3),IATST
1044	2 CONTINUE
1045	RETURN
1046	1 CALL LEOR(256,IOUT1,IOUT1)
1047	RETURN
1048	3 CALL LA(MASK13,ICON2,IMNR)
1049	CALL LEOR(IMNR,ICON2,ICON2)
1050	ICORD = ICORD + IMNR
1051	IMINF = 1
1052	GO TO 1
1053	END
1054	BLOCK DATA
1055	COMMON /EI/ IN,TA,TE,TX,TY,TD,TP,TT,TS,TR,TM
1056	DATA IN,TA,TE/SHN ,SHA ,SHE /
1057	DATA TX,TY,TD,TP/SHX ,SHY ,SHD ,SHP
1058	DATA TT,TS,TR,TM/SHT ,SHS ,SHR ,SHM
1059	END
1060	SUBROUTINE SETM (I)
1061	
1062	THIS IS AN ASSEMBLY LANGUAGE ROUTINE

CARD NO	INPUT LISTING	AUTOFLOW CHART SET -	PPSS
07/22/71			
CARD NO	*****	CONTENTS	*****
1063	C		
1064	RETURN		
1065	END		
1066	SUBROUTINE SETC (I)		
1067	C		
1068	C	THIS IS AN ASSEMBLY LANGUAGE ROUTINE	
1069	C		
1070	RETURN		
1071	END		
1072	SUBROUTINE LN (I,J,K)		
1073	C		
1074	C	THIS IS AN ASSEMBLY LANGUAGE ROUTINE	
1075	C		
1076	RETURN		
1077	END		
1078	SUBROUTINE LENR (I,J,K)		
1079	C		
1080	C	THIS IS AN ASSEMBLY LANGUAGE ROUTINE	
1081	C		
1082	RETURN		
1083	END		
1084	SUBROUTINE LS (I,J,K)		
1085	C		
1086	C	THIS IS AN ASSEMBLY LANGUAGE ROUTINE	
1087	C		
1088	RETURN		
1089	END		
1090	SUBROUTINE RS (I,J,K)		

1091	C	
1092	C	THIS IS AN ASSEMBLY LANGUAGE ROUTINE
1093	C	
1094		RETURN
1095		END
1096		SUBROUTINE NGFRW (I,J,K,L,M)
1097	C	
1098	C	THIS IS AN ASSEMBLY LANGUAGE ROUTINE
1099	C	
1100		RETURN
1101		END
1102		SUBROUTINE RGEF (I)
1103	C	
1104	C	THIS IS AN ASSEMBLY LANGUAGE ROUTINE
1105	C	
1106		RETURN
1107		END
1108		SUBROUTINE TIME(I,J,K)
1109	C	THIS IS A SYSTEM SUBROUTINE
1110		RETURN
1111		END
1112		SUBROUTINE LA(I,J,K)
1113	C	THIS IS AN ASSEMBLY LANGUAGE SUBROUTINE
1114		RETURN
1115		END

07/22/71

PROCEDURAL STATEMENT LABEL INDEX

AUTOFLOW CHART SET -

RPSS

PG.BX	NAME	PG.BX	NAME	PG.BX	NAME	PG.BX	NAME	PG.BX	NAME
19.07	GNP	26.01	SRCH	20.13	2	6.07	310	9.17	401
34.01	IMFSC	54.02	TIME	32.07	2	6.06	311	10.01	402
28.01	INTWDL	32.05	1	34.20	2	6.05	312	10.03	403
55.02	LA	26.02	1	42.04	2	7.01	313	10.06	404
49.02	LFUR	42.06	1	26.09	2	7.05	314	10.13	405
48.02	LN	2.14	1	3.21	200	5.17	315	10.24	406
50.02	LS	22.06	1	3.19	21	8.12	316	19.16	5
52.02	MGFRW	19.14	1	24.12	3	8.31	317	24.04	5
16.01	MMENT	40.07	1	19.10	3	6.21	318	3.13	50
36.01	MPAGE	24.09	1	42.08	3	6.24	319	2.22	500
40.01	MSET	34.12	1	26.11	3	7.02	320	17.18	6
30.01	OUTWDL	16.14	1	32.09	3	8.17	350	19.01	6
32.01	PBYTE	13.19	10	4.08	300	5.14	360	24.02	6
38.01	PWMSTT	14.01	100	4.12	301	7.23	370	8.05	600
53.02	RECF	13.10	11	4.18	302	7.04	380	20.03	7
51.02	RS	14.18	14	5.09	303	4.31	390	17.16	7
27.01	SCAN	14.13	15	5.03	304	8.16	391	3.08	700
13.01	SCUDE	14.23	16	4.12	305	24.13	4	9.04	7000
42.01	SFTA	24.16	2	5.01	306	19.19	4	3.11	701
47.02	SFTC	13.06	2	6.20	307	14.19	4	17.23	8
46.01	SFTM	17.01	2	6.09	308	26.07	4	12.09	9
24.08	SHLSRT	40.05	2	6.01	309	5.01	400	9.24	999

07/22/71 TABLE OF CONTENTS AND REFERENCES
 CARD TO PAGE/BOX NAME
 REFERENCES (SOURCE SEQUENCE NO. AND PAGE/BOX)

FORTRAN MODULE RPSS

CHART TITLE - INTRODUCTORY COMMENTS

CHART TITLE - PROCEDURES

(000075)	2.14 1	(000043)	2.18	(000402)	10.09				
(000094)	2.22 500	(000403)	10.09						
(000111)	3.08 700								
(000111)	3.08	(000111)	3.09						
(000113)	3.11 701								
(000113)	3.11	(000113)	3.12						
(000114)	3.13 50	(000119)	3.17						
(000122)	3.19 21	(000119)	3.16						
(000124)	3.21 200	(000121)	3.18	(000397)	10.25				
(000299)	4.01 390	(000145)	4.11						
(000143)	4.08 300	(000153)	4.13	(000147)	4.17	(000179)	5.01	(000194)	5.11
		(000246)	7.03						(000243) 7.01
(000144)	4.10	(000143)	4.08						
(000152)	4.12 301	(000144)	4.10						
(000160)	4.18 302	(000153)	4.12						
(000179)	5.01 306	(000175)	4.23						
(000144)	5.03 304	(000201)	5.13	(000206)	5.15				
(000194)	5.09 303	(000175)	4.23						
(000205)	5.14 360	(000165)	4.19						
(000232)	6.01 309	(000225)	6.10	(000274)	7.14				
(000233)	6.07	(000238)	6.06						

(000237)	6.05	312	(000234)	6.03	
(000238)	6.06	311			
(000240)	6.07	310	(000228)	6.11	
(000221)	6.09	308	(000218)	6.19	
(000210)	6.12	305	(000190)	5.09	(000297) 8.15
(000211)	6.14		(000210)	6.12	
(000217)	6.17	315	(000259)	6.27	
(000250)	6.20	307	(000212)	6.15	
(000253)	6.21	318	(000303)	4.04	(000267) 7.04
(000257)	6.24	319			
(000257)	6.24		(000257)	6.25	
(000243)	7.01	313	(000241)	6.08	
(000244)	7.02	320	(000293)	8.13	
(000245)	7.03	370	(000156)	4.15	
(000261)	7.04	380	(000156)	4.14	
(000263)	7.05	314	(000241)	6.08	
(000268)	7.11		(000266)	7.08	
(000275)	8.01	317	(000241)	6.08	
(000282)	8.05	600	(000279)	8.07	
(000292)	8.12	316	(000269)	7.11	(000269) 7.11
(000305)	8.16	391	(000292)	8.12	
(000309)	9.17	350			
(000313)	9.01	400	(000305)	8.16	
(000319)	9.04	7000	(000316)	9.02	
(000343)	9.17	401	(000331)	9.11	
(000407)	9.24	999	(000358)	9.23	
(000363)	10.01	402	(000333)	9.15	
(000368)	10.03	413	(000350)	9.23	

07/22/71 TABLE OF CONTENTS AND REFERENCES AUTOFLOW CHART SET -
 CARD ID PAGE/BOX NAME REFERENCES (SOURCE, SEQUENCE NO., AND PAGE/BOX)

(000398) 10.05 404 (000369) 10.04 (000377) 10.10
 (000389) 10.13 405 (000381) 10.11 (000391) 10.17
 (000396) 10.24 406 (000385) 10.12

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE SCODE

(000416) 13.01 SCODE (000122) 3.19-X
 (000449) 13.06 2
 (000449) 13.06 (000449) 13.07
 (000458) 13.09 9 (000466) 13.15 (000533) 14.22
 (000461) 13.10 11 (000483) 13.21
 (000481) 13.19 10 (000482) 13.29 (000494) 13.22 (000487) 13.24
 (000487) 13.26 (000486) 13.24
 (000491) 14.01 100 (000464) 13.13 (000495) 13.23
 (000501) 14.06 (000511) 14.13
 (000511) 14.13 14 (000507) 14.12
 (000521) 14.18 14 (000515) 14.14 (000535) 14.24
 (000534) 14.23 14 (000529) 14.21

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE NAME (IFLG)

(000543) 16.01 NAME (000481) 13.19-X
 (000540) 16.13 (000589) 14.11
 (000572) 16.14 1 (000545) 14.04

(000649)	16.19	4	(000577)	16.08	(000572)	16.15	(000574)	16.15	(000575)	16.17	(000576)	16.18
(000594)	17.01	2	(000583)	16.09								
(000597)	17.04		(000595)	17.02								
(000602)	17.07		(000600)	17.05								
(000608)	17.10		(000606)	17.08								
(000611)	17.11		(000633)	17.18								
(000625)	17.16	7	(000616)	17.13								
(000633)	17.19	6										
(000636)	17.21		(000634)	17.19								
(000641)	17.23	8	(000629)	17.17								

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE GNR

(000727)	19.01	6	(000674)	19.13	(000712)	19.25
(000729)	19.04		(000728)	19.02		
(000661)	19.07	GNR	(000714)	4.00-X		
(000673)	19.10	3	(000695)	19.14		
(000695)	19.14	1	(000648)	19.07		
(000695)	19.14		(000693)	20.11		
(000697)	19.16	5	(000714)	19.26		
(000704)	19.19	4	(000696)	19.15		
(000712)	19.25		(000711)	19.22		
(000680)	20.03	7	(000721)	20.21		
(000685)	20.07		(000683)	20.04		
(000693)	20.11		(000692)	20.09		
(000716)	20.13	2	(000677)	20.01		

AUTOFLOW CHART SET -
REFERENCES (SOURCE SEQUENCE NO. AND PAGE/BOX)

07/22/71 TABLE OF CONTENTS AND REFERENCES

CARD NO	PAGE/BOX	NAME
(000719)	20.16	(000718) 20.14
(000720)	20.18	(000719) 20.16
(000721)	20.21	(000720) 20.18

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE SCANIVAL(K)

(000735)	22.01	SCAP	(000612) 17.12-X
(000742)	22.03		(000748) 22.06
(000749)	22.06	1	(000742) 22.03

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE SHLSRT

(000777)	24.01		(000779) 24.03
(000779)	24.02	6	
(000785)	24.06	5	(000772) 24.14
(000757)	24.08	SHLSRT	(000491) 14.01-Y
(000763)	24.09	1	(000789) 24.07
(000770)	24.12	3	(000789) 24.07
(000771)	24.13	4	(000781) 24.05
(000793)	24.16	2	(000764) 24.10

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE SACH(IARG,K)

(000799)	26.01	SRCH	(000174)	4.22-X	
(000805)	26.02	1	(000813)	26.06	(000818) 26.12
(000827)	26.07	4	(000805)	26.02	
(000822)	26.09	2	(000807)	26.04	
(000814)	26.11	3	(000809)	26.05	

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE INTWDLK)

(000832)	28.01	INTWDL	(000449)	13.06-X	(000621) 17.16-X
----------	-------	--------	----------	---------	------------------

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE OUTWDLK)

(000847)	30.01	OUTWDL	(000523)	14.04-X	
----------	-------	--------	----------	---------	--

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE PAYTE(IM,IR,IM,INR)					
(000899)	32.01	PAYTE	(000547)	2.09-X	
			(000258)	6.26-X	
(000904)	32.05	1			
(000906)	32.07	2	(000903)	32.04	(000909) 32.10
(000908)	32.09	3	(000903)	32.04	
			(000221)	5.09-X	(000257) 5.24-X
			(000255)	4.22-X	

07/22/71 TABLE OF CONTENTS AND REFERENCES AUTOFLD CHART SFT -
 CARD 17 PAGE/BOX NAME REFERENCES (SOURCE SEQUENCE NO. AND PAGE/BOX)

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE	IMFSG		
(000013)	34.01 IMFSG	(000120)	3.22-X
(000036)	34.08	(000034)	34.06
(000047)	34.12 1	(000038)	34.09
(000059)	34.20 2	(000054)	34.17
		(000146)	4.16-X
		(000050)	34.16

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE NPAGE

(000064)	34.01 NPAGE	(000494)	14.07-X
		(000510)	14.12-X

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE PRNSTT

(000095)	34.01 PRNSTT	(000322)	9.06-X
----------	--------------	----------	--------

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE NSET

(001015)	40.01 NSET	(000129)	3.23-X
(001024)	40.03	(001027)	40.07
(001025)	40.04	(001026)	40.06
		(000244)	7.02-X
		(000294)	8.14-X

(001026) 40.05 2
(001027) 40.07 1

LIST TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE SETA

(001032)	42.01	SETA	(000210)	6.13-X	
(001046)	42.04	2	(001042)	42.02	(001043) 42.03
(001048)	42.06	1	(001043)	42.03	(001052) 42.10
(001048)	42.08	3	(001043)	42.03	

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - BLOCK DATA

CHART TITLE - NON-PROCEDURAL STATEMENTS

CHART TITLE - SUBROUTINE SETM(I)

(001062) 46.01 SETM (000062) 2.07-X

CHART TITLE - SUBROUTINE SETC(I)

(001068) 47.02 SETC (000063) 2.08-X

CHART TITLE - SUBROUTINE LO(I,J,K)

07/22/71	TABLE OF CONTENTS AND REFERENCES			AUTOFLOW CHART SET -			PAGE
CARD 17	PAGE/POX	NAME	REFERENCES	(SOURCE SEQUENCE NO. AND PAGE/POX)			
(001074)	48.02	LO	(000134)	3.26-X (000455)	28.05-X (000906)	32.07-X	
CHART TITLE - SUBROUTINE LEOR(I,J,K)							
(001080)	49.02	LEOR	(000135)	3.27-X (000854)	28.06-X (001044)	42.09-X (001049)	
CHART TITLE - SUBROUTINE LS(I,J,K)							
(001086)	50.02	LS	(000070) (000908)	2.12-X 32.09-X	3.25-X (000133)	4.05-X (000136)	28.07-X (000857)
CHART TITLE - SUBROUTINE RS(I,J,K)							
(001092)	51.02	RS	(000069) (000905)	2.11-X 32.06-X	4.21-X (000170)	5.07-X (000188)	30.07-X (000884)
CHART TITLE - SUBROUTINE MGFRW(I,J,K,L,M)							
(001098)	52.02	MGFRW	(000264)	7.06-X (000344)	9.18-X (000363)	10.01-X (000674)	19.20-X (000709)
CHART TITLE - SUBROUTINE REOF(I)							
(001104)	53.02	REOF	(000672)	19.09-X			
CHART TITLE - SUBROUTINE TIME(I,J,K)							
(001109)	54.02	TIME	(000141)	4.07-X (000290)	8.11-X		
CHART TITLE - SUBROUTINE LA(I,J,K)							
(001113)	55.02	LA	(000169) (000205) (001048)	4.20-X 5.14-X 42.08-X	5.04-X (000185) 30.03-X (000880)	5.04-X (000187) 30.05-X (000882)	5.12-X (000200) 32.07-X (000901)

07/22/71

AUTOFLOW CHART SET -

RPSS

PAGE 01

CHART TITLE - INTRODUCTORY COMMENTS

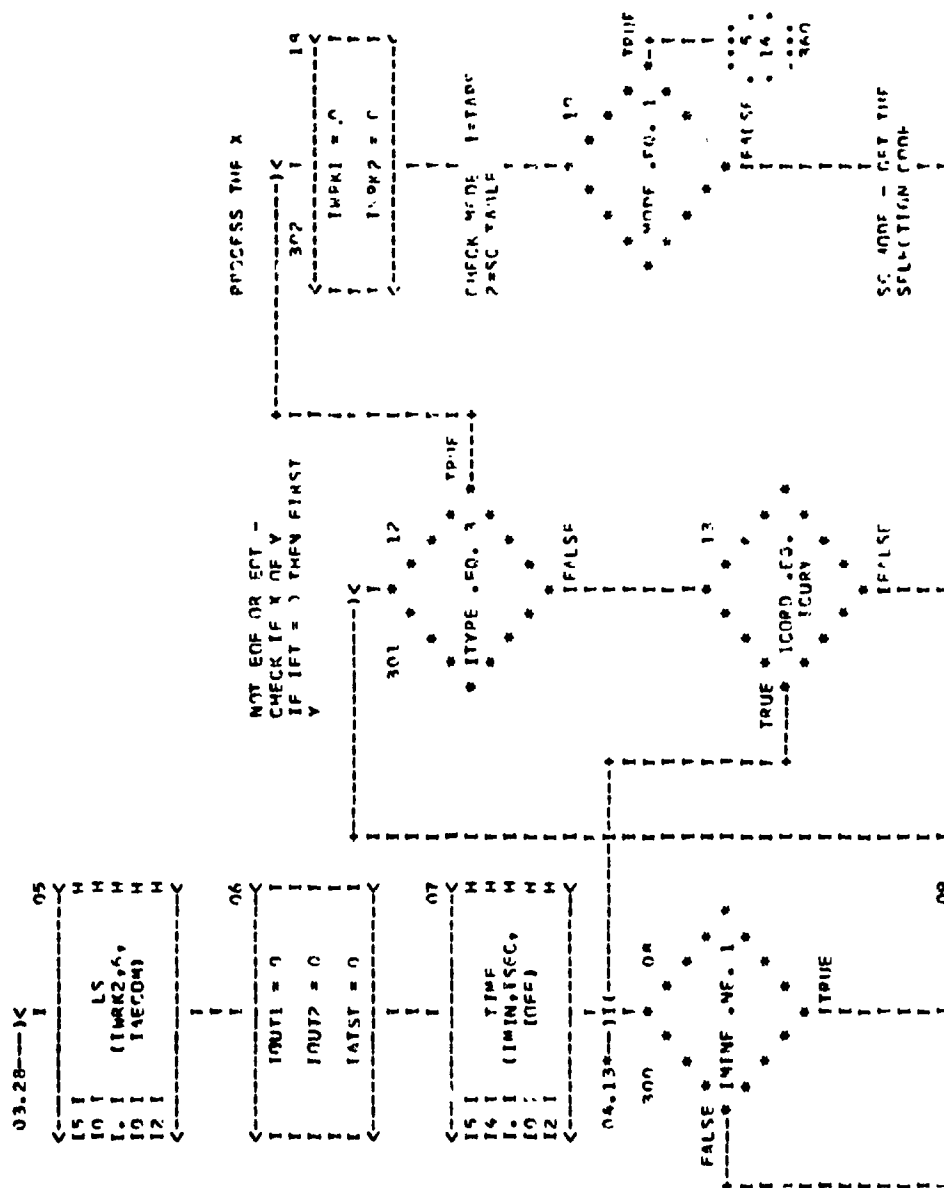
INITIALIZATION

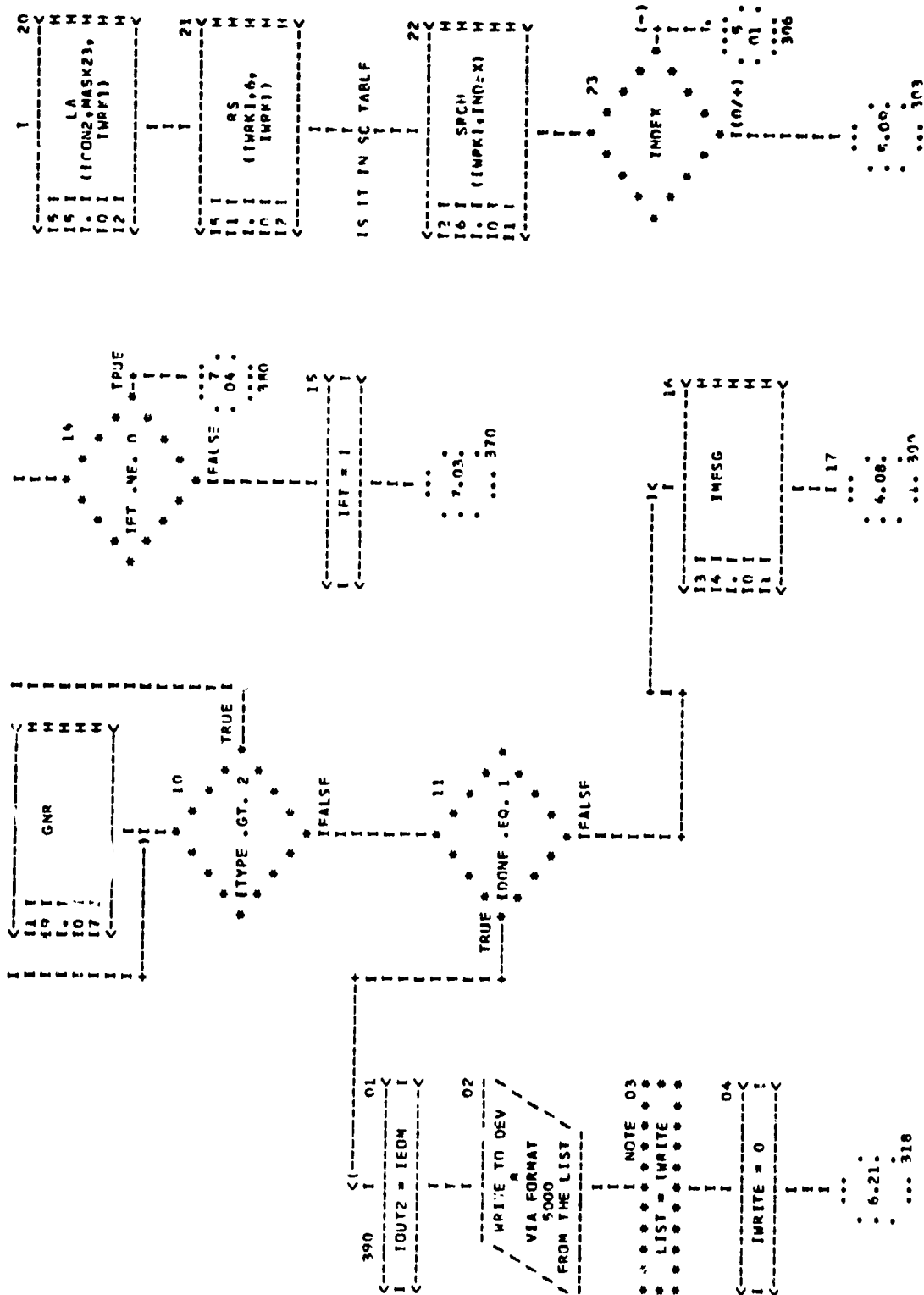
Preceding page blank

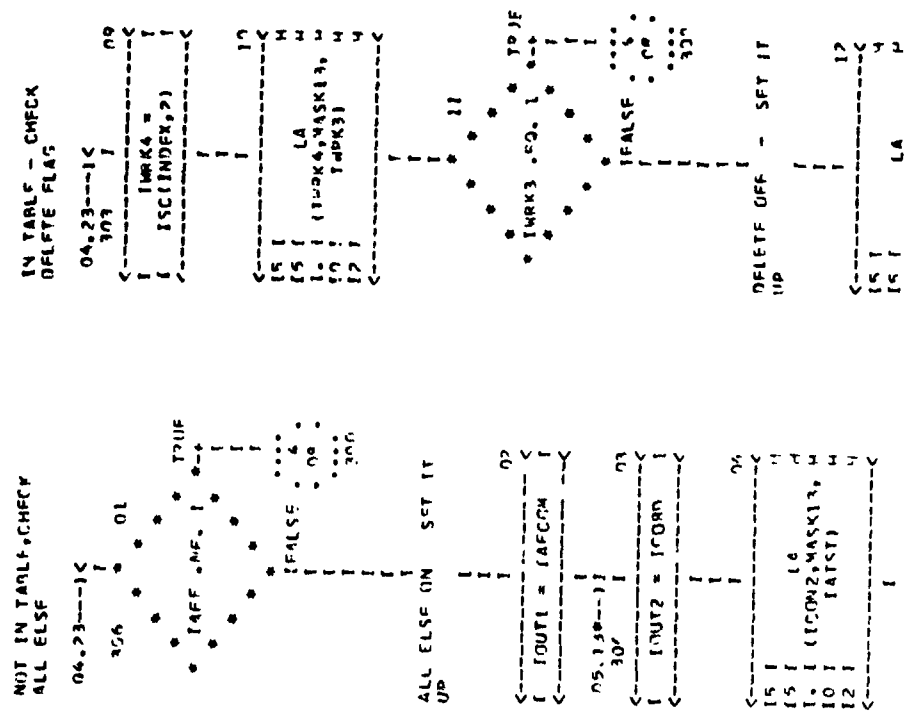


100-100000

SECRET








```

I- I (IMRK4, MASK12, M
I0 I (OUT1) M
I2 I M
<----->
I
I 13
...
. 5.03.
. .... 304

```

```

TAPE MODE
04.19---><
360 I 14
<----->
I5 I 4
I5 I LA
I- I (ICON2, MASK12, M
I0 I (OUT1) M
I2 I M
<----->
I
I 15
...
. 5.03.
. .... 304

```

```

I I 05
<----->
I (CNT(IATST) = I
I (CNT(IATST) + 1) I
<----->
I
I
I 06
<----->
I5 I LA
I5 I (OUT1, 100,
I- I (IMRKCH)
I0 I M
I2 I M
<----->
I
I
I 07
<----->
I5 I PS
I1 I (IMRKCH, 6,
I- I (IMRKCH)
I0 I M
I2 I M
<----->
I
I
I 08
<----->
I (CH(IMRKCH + 1) = I
I (CH(IMRKCH + 1) + I
I I
I I
<----->
I
I
I
...
. 6.12.
. .... 305

```

CHART TITLE - PROCEDURES

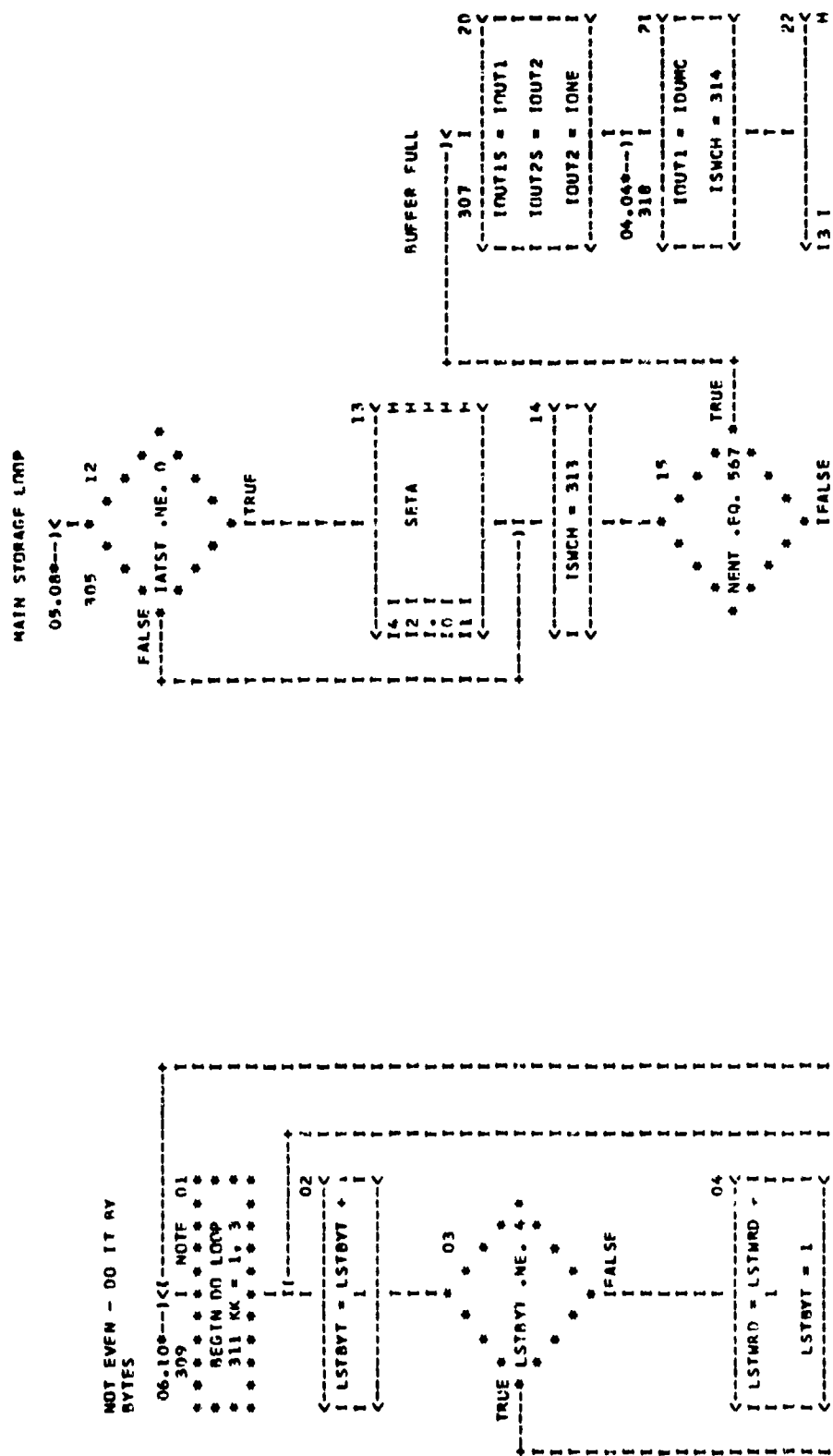


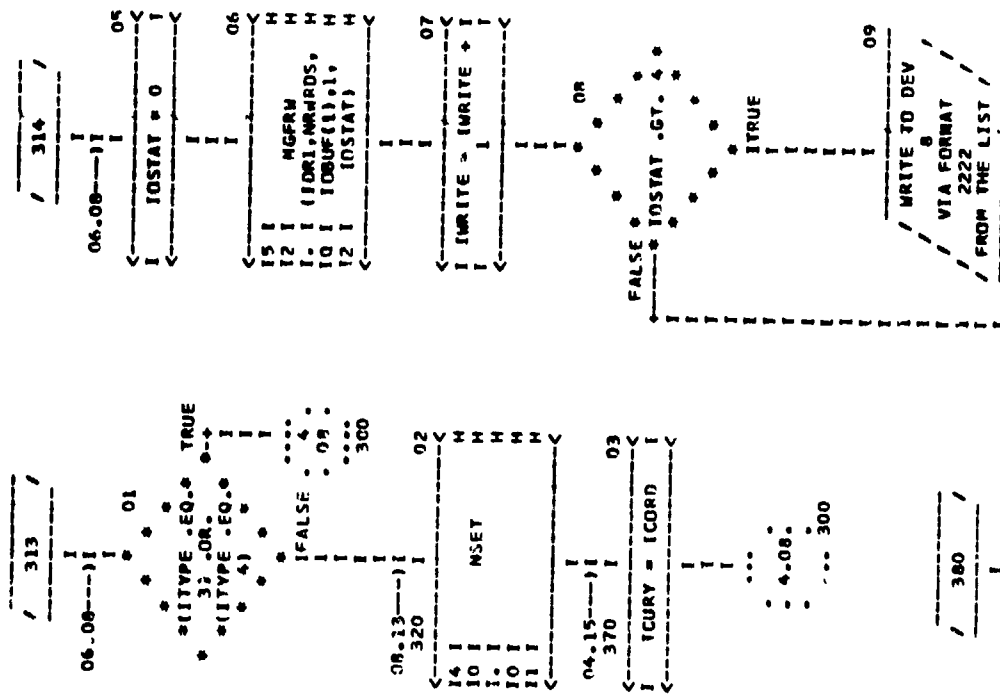


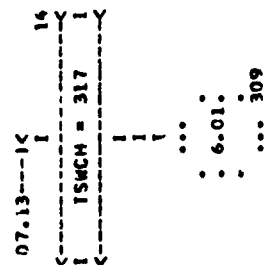
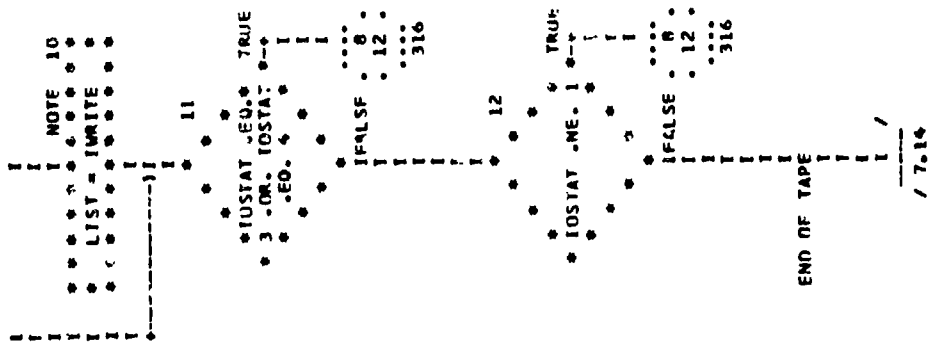
CHART TITLE - PROCEDURES

AUTOFLOW CHART SET

RPSS

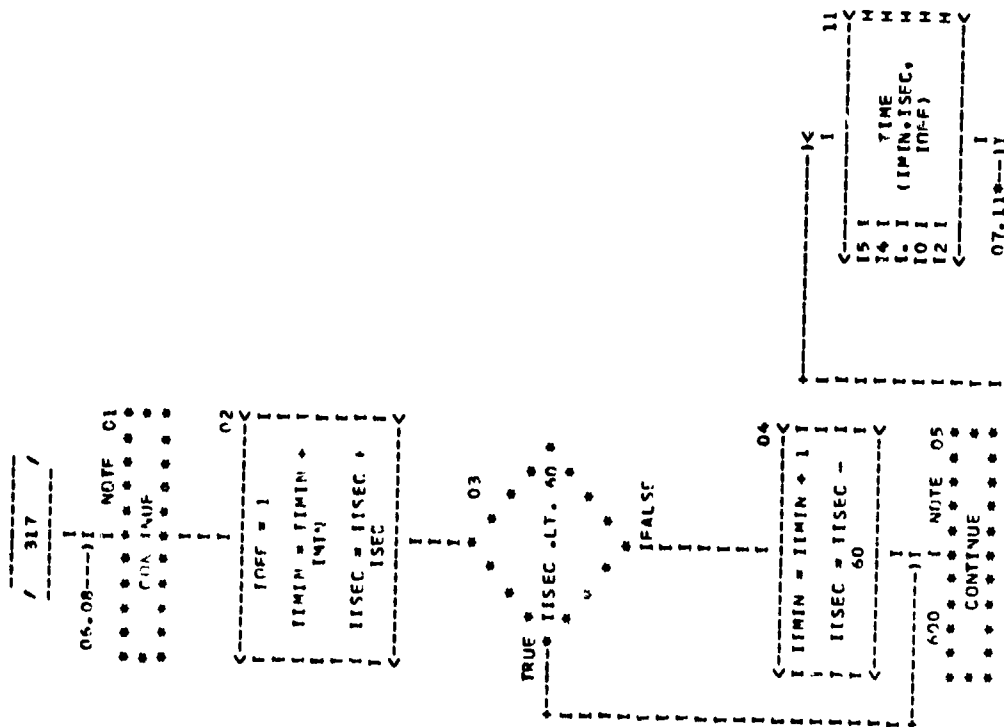
PAGE 07





PAGE 08

CHART TITLE - PROCEDURES



OUTPUT TAPF ERROR

390 1 17
MALT

21. 2000; 10: 545-550



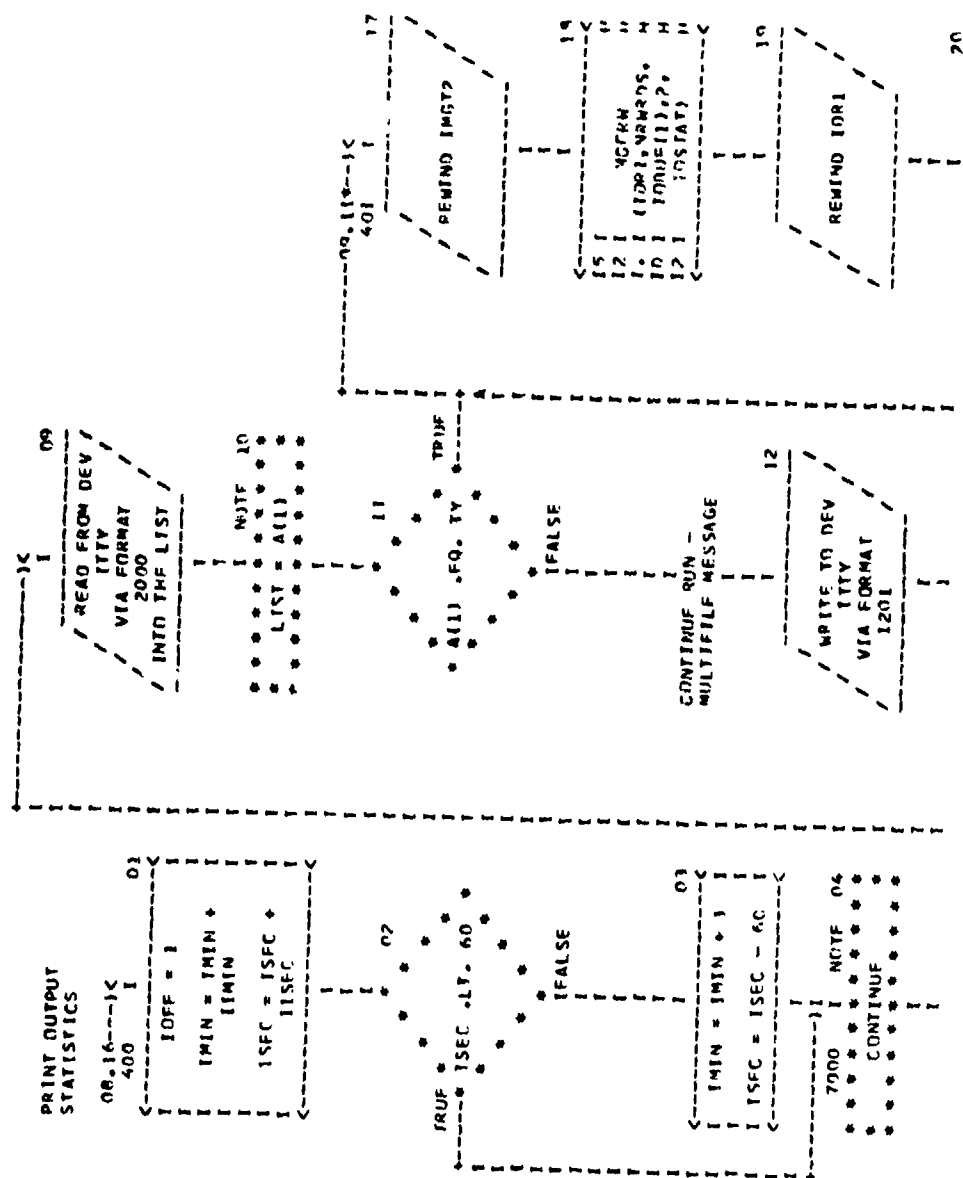
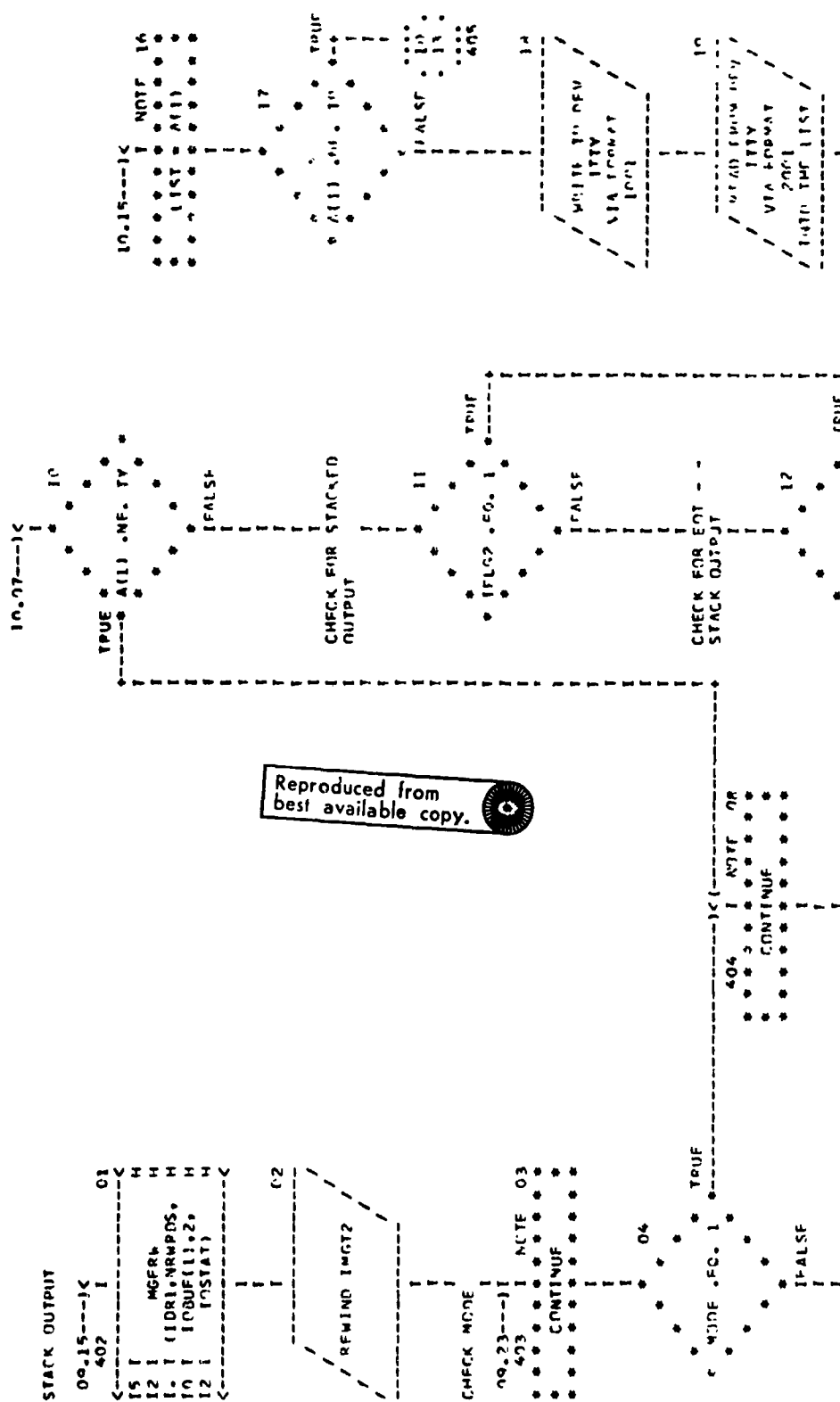




CHART TITLE - PROCEDURES



07/22/71

AUTOFLOW CHART SET -

APSS

PAGE 11

CHART TITLE - NON-PROCEDURAL STATEMENTS

```

DIMENSION TSC(1000,2),A(9)
DIMENSION MASK(3),MASK2(3),MASKS(4)
DIMENSION ITR<(1743)
DIMENSION INSET(4),INOUT(1364)
DIMENSION ICNT(24),ICH(4)
COMMON /A1/ TSC,ITMAX,ITCNR,MENSC,MAXSC
COMMON /B1/ TNSC,INCH,INDCN,INDCL
COMMON /C1/ IAF,IAFCH,IAFCHN
COMMON /D1/ A,I N,TSCS,ISCF,EDDEN
COMMON /E1/ TN,TA,TF,TX,TY,TD,TP,TT,TS,TP,TM
COMMON /F1/ ITTY,TD,INT,INP,IPRT,
COMMON /G1/ IORL,IOR2,IMP3,INFC
COMMON /H1/ INSC,INCH,INDCN,INDCL
COMMON /I1/ I-MINTM,IDAY,IYEAR,IPAGE,ITILF,IFEV
COMMON /J1/ IMGT1,IMGT2,INDNE
COMMON /MASK/ MASK1,MASK2
COMMON /GNR1/ ITYPE,ICNDR,ICN1,ICN2,NK,LX,IFT,ISTAT
COMMON /TRUF/ ITRUF,NYS,NXS
COMMON /AFIC/ IAFIC,INAP
COMMON /INUT/ INUT1,INUT2,ISTAT
COMMON /K1/ ICNT,ICM
COMMON /L1/ ICHN,ISCF,IDFF
COMMON /M1/ INSET,LISTAP,LISTBYT,NFNT,INOUT
EQUIVALENCE (MASK1(1),MASKS(1))
EQUIVALENCE (MASK1(1),MASK1(2)),(MASK1(2),MASK1(3))

```

Reproduced from
best available copy.



1000	EQUIVALENCE (MASK21,MASK2(11)),(MASK22,MASK2(21)),(MASK23,MASK2(31)) FORMAT (28H MOUNT NEW SCRATCH ON DRIVE ,11,19H ENTER 0 WHEN SH DONE)
2000	FORMAT (A1)
1001	FORMAT (46H ENTER MONTH,DAY,YEAR AS 6 NUMERICS ... M000YY)
2001	FORMAT (312)
1002	FORMAT (29H ENTER 5 DIGIT TAPE ID NUMBER)
2002	FORMAT (15)
1003	FORMAT (39H MOUNT 1ST REEL OF INPUT TAPE ON DRIVE ,11)
1101	FORMAT (49H ENTER 6 DIGIT MINIMUM AREA IN PROPER RESOLUTION.
2100	FORMAT (16)
1100	FORMAT (50H DO YOU WISH TO SUPPLY SELECTION CODES EXTERNALLY / 13H ENTER Y OR N)
2222	FORMAT (23H BAD WRITE AT IWRITE = ,5-)
1050	FORMAT (40H AT PAUSE 2, MOUNT NEW SCATCH ON DRIVE ,11,14H END OF MINUTE)
5000	FORMAT (16H TOTAL WRITES = ,15,4H +)
1200	FORMAT (42H DO YOU WISH TO TERMINATE RUN (Y OR N))
1201	FORMAT (47H DO YOU WISH TO STACK OUTPUT FILES (Y OR N))
1202	FORMAT (24H REMOVE OUTPUT TAPE ON DRIVE ,1X,11/ 15H LABEL IT DATE, 4X,12,1H/,12,1H/,12,14H TAPE NUMBER ,16,14H REEL NUMBER ,12)
1203	FORMAT (61H DO YOU WISH TO USE THE SAME SELECTION CODES AS PREVIOUS

07/22/71

AUTOFLOW CHART SET -

RPSS

PAGE 12

CHART TITLE - NON-PROCEDURAL STATEMENTS

1204 S RUN / 11H (Y OR N))
FORMAT (42H END OF RUN - REMOVE ALL TAPES FROM DRIVES)///20X,20H **
* TERMINATION ***)

CHART TITLE - NON-PROCEDURAL STATEMENTS

```

1004 DIMENSION ISC(1000,2),A(9)
      COMMON /A1/ ISC,ITMAX,ITCUR,MINSC,MAXSC
      COMMON /B1/ INSC,INCH,INDEF,INDEL
      COMMON /C1/ IAEF,IAFCH,IAFDEN
      COMMON /D1/ A>IDEN,ISCS,ISCF,IODEFN
      COMMON /E1/ TH,TA,TF,TX,YY,TD,TP,TT,TS,TR,TM
      COMMON /F1/ ITTY,IPT,INT,IMP,IPATR
      COMMON /G1/ IOM1,IMR2,IMR3,IOEC
      COMMON /H1/ INSC,IOCH,IODEFN,INDEL
      COMMON /I1/ IMONTH,IDAY,IYEAR,IPAGE,IFILE,IREV
      FORMAT (45H ENTER DEFAULT DENSITY AS NUMERIC FROM 0 TO 7)
2003 FORMAT (11)
1009 FORMAT (51H ENTER CHANNEL NUMBER FROM 1 TO 4 OR ENTER 0 TO END)
      H SELECTION CODE INPUT EACH TIME CH NR.= IS TYPED)
      FORMAT (8H CH NR =)
1011 FORMAT (12H ENTER COOFS,2X,15H(WYZNNNNNNNNNN))
1012 FORMAT (6X,15,15X (2,11X,11,12X,11)
1018 FORMAT (5X,6HALL ELSE,13X,12,11X,11)
1019 FORMAT (61H SELECTION CODE SUMMARY ON PRINTER. ARE COOFS, I,IMP,TS,TR,
1020 UIRED / 14H ENTER Y OR N.)

```

07/22/71

CHART TITLE - SUBROUTINE MKENT(IFLG)

AUTOFLOW CHART SET -

2PSS

PAGE 14

/ NEXT /

13.19----->

*** READS THE INPUT
SELECTION CODE AND
ENTERS IT INTO THE
*** SELECTION CODE
TABLE

01
IFLG = 0
----->

*** SETS THE ENTRY

02
FROM FROM DEG /
14P
VIA FORMAT /
2004
INTO THE LIST /

03
NOTE 03
LIST = A(1),
A(2), IDFN, TSCS,
ISCF

*** TEST FOR END

04
ALL .NE. TX
TRUE

*** NOT END --
CHECK THE ENTRY
*** FOR CHECK CODE

14.04----->
NOTE 14
ALL .NE. TA
AND. (ALL) .NE.
TE) .AND. (ALL)
NE. TX) .AND.
(ALL) .NE. TN

05
SEE NOTE
ANDVE

IFALSE

06
IA(2) .NE.
TS) .AND.
IA(2) .NE.
TR)

IFALSE

*** ENTRY OK --- NOW
DETERMINE THE TYPE

*** FOR THE FOUND -
POINT FOR THE MESSAGE

14.08----->
NOTE 14
NOTE TO VCV /
-TV
VIA FORMAT /
100P
FROM THE LIST /

07
NOTE 20
LIST = A(1),
A(2), IDFN, TSCS,
ISCF

121





07/22/71

AUTOFLOW CHART SET -

RPSS

PAGE 14

CHART TITLE - NON-PROCEDURAL STATEMENTS

```

2004 DIMENSION ISC(1000,2), AI(9)
1012 COMMON /A1/ ISC,ITMAX,ITCUR,NINSC,MAXSC
1010 COMMON /B1/ INSC,INCH,INDEN,INDEL
1008 COMMON /C1/ IAEF,IAECH,IAEDEN
COMMON /D1/ A,IDEN,ISCS,ISCE,INDEN
COMMON /E1/ TN,TA,TE,TX,TY,TD,TP,TT,TS,TR,TN
COMMON /F1/ ITTY,IPT,ITD,TNP,IPRTR
COMMON /G1/ IOR1,IOR2,ION3,IDEC
COMMON /H1/ INSC,INCH,INDEN,INDEL
COMMON /I1,AL,IL,215)
COMMON /J1,OK)
COMMON /K1 SELECTION CODE TABLE FULL - LAST ENTRY,
COMMON /L1,AL,IL,IX,IS,IX,IS)
COMMON /M1 SELECTION CODE ENTRY - WILL NOT BE PUT IN TA
COMMON /N1 ENTRY AS RECEIVED IS /IX,AL-IX,AL,IX,IL,IX,IS,IX,TS/43H A
COMMON /O1 CONTINUE OR MANUALLY TERMINATE)

```


07/22/71

CHART TITLE - SURROUTLINE GNR

A TOFLOW CHART SET -

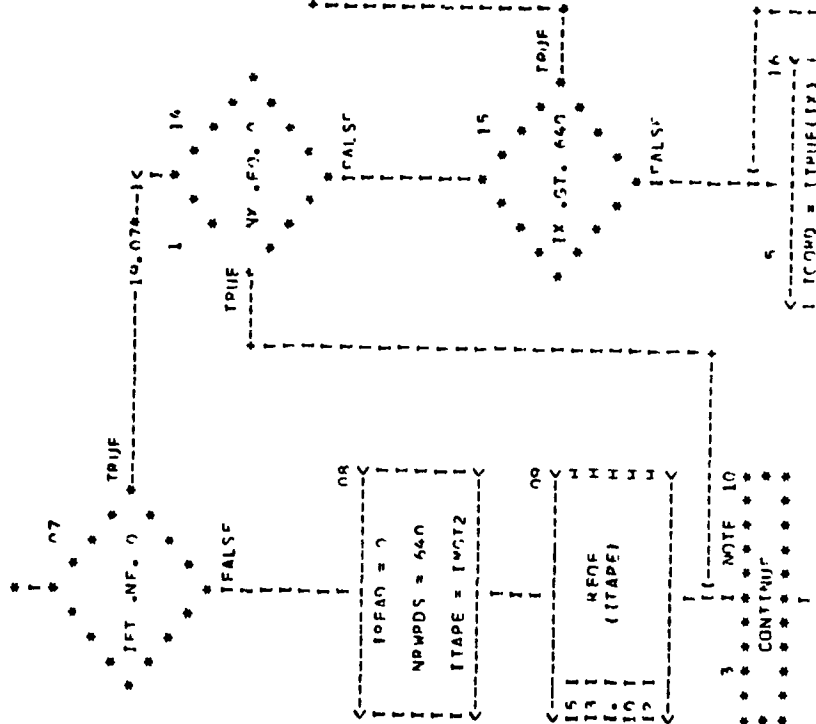
PPSS

NAME 10

GNP /

04.09---1<

*** GET THE NEXT
RECORD FROM INPUT
TAPE



Preceding page blank

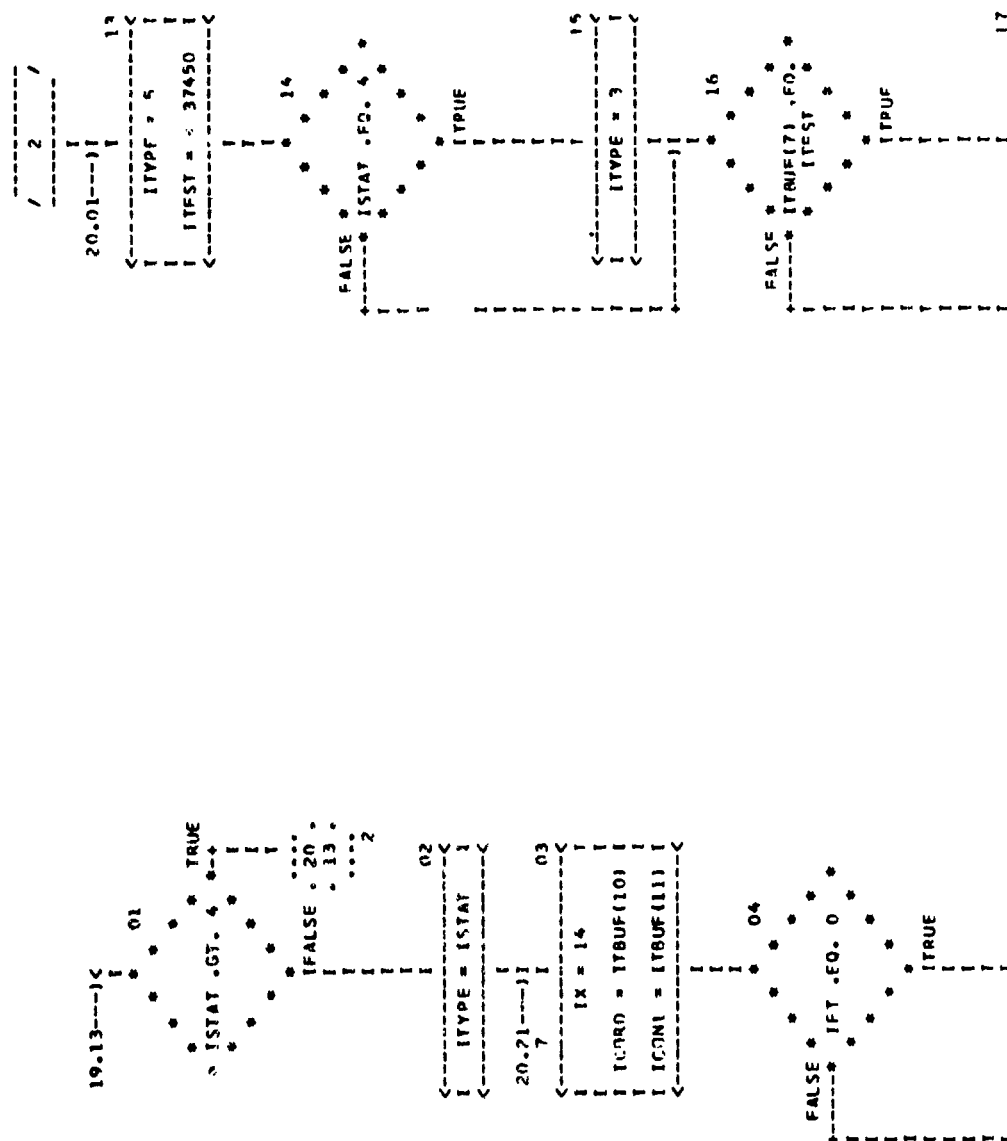
07/22/71

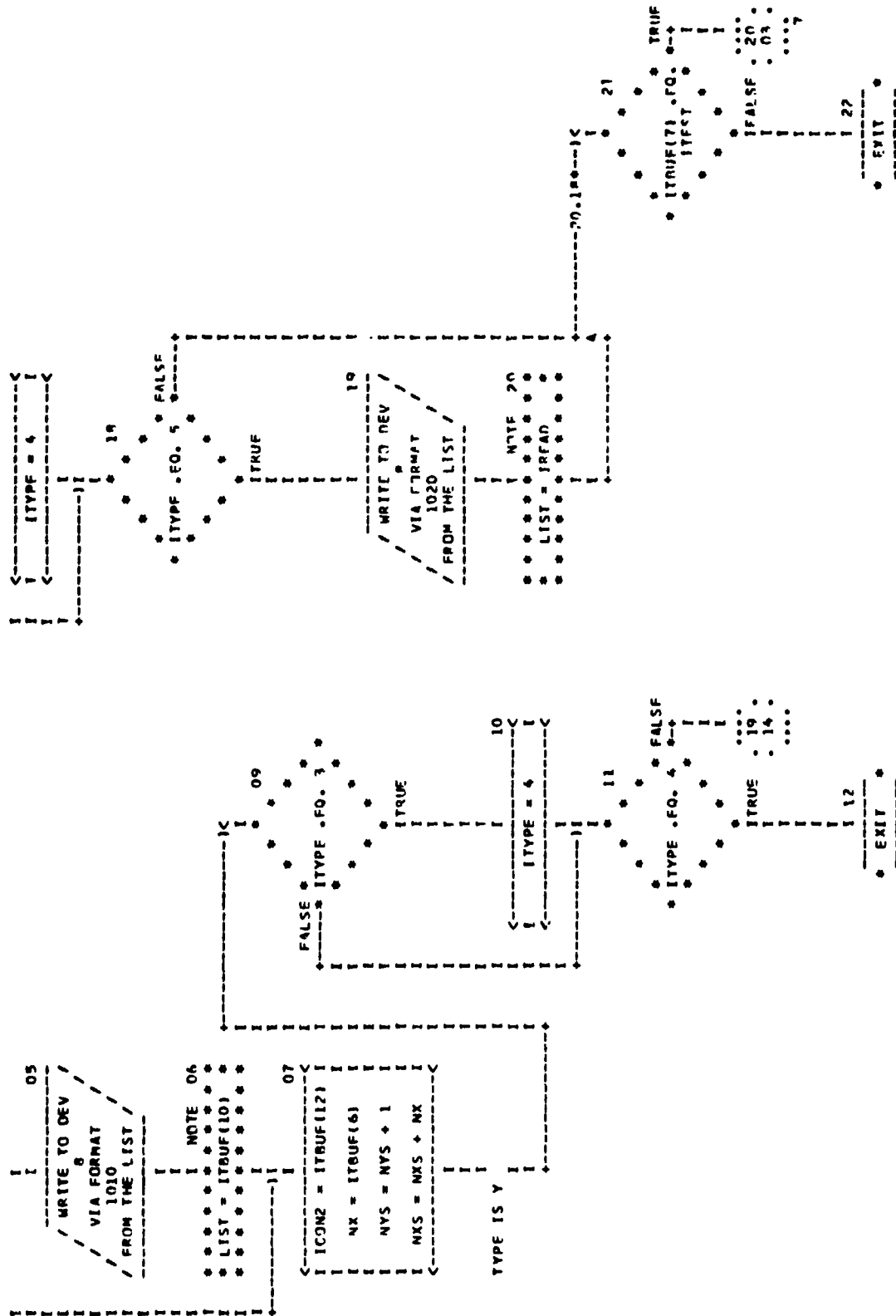
CHART TITLE - SUBROUTINE GNR

AUTOFLOW CHART SET -

RPSS

PAGE 20





07/22/71

CHART TITLE - NON-PROCEDURAL STATEMENTS

```

DIMENSION ITRUF(740)
COMMON /GNR1/ ITYPE,ICORD,ICON1,ICON2,NX,IX,IFT,ISTAT
COMMON /TRUF/ ITRUF,NYS,NYS
COMMON/J1/IMG1,IMG2,IMG3
1010 FORMAT (23H FIRST Y COORDINATE IS ,I6)
1020 FORMAT (21H RAD READ AT IPEAD = ,I5)
1030 FORMAT (15H TOTAL READS = ,I5)
    
```

Preceding page blank



$\begin{array}{|c|} \hline K = -1 \\ \hline \end{array}$
 $\begin{array}{c} \text{I} \\ \text{I OR} \\ \text{I} \end{array}$
 $\begin{array}{|c|} \hline \text{FXIV} \\ \hline \end{array}$

$\begin{array}{|c|} \hline \text{FXIV} \\ \hline \end{array}$

07/22/71

AUTOFLOW CHART SET -

EPSS

PAGE 23

CHART TITLE - NON-PROCEDURAL STATEMENTS

DIMENSION ISF(1000,2)

COMMON /A/ ISF,ITMAX,ITCUR,NINSC,MAXSC

CHARITY T. LEE - SUBROUTINE SHLSRT

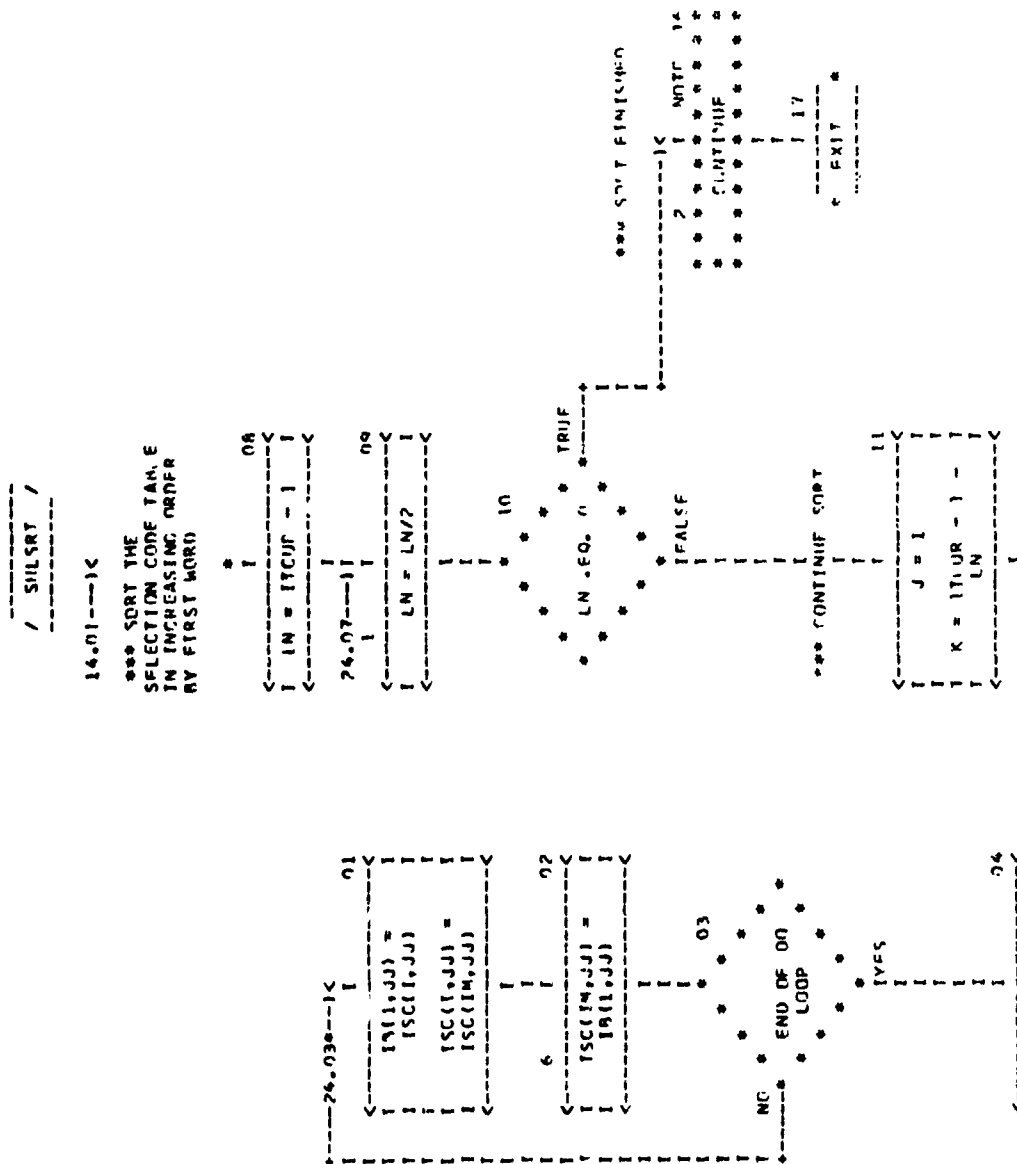
AUTOFLOW CHART SET -

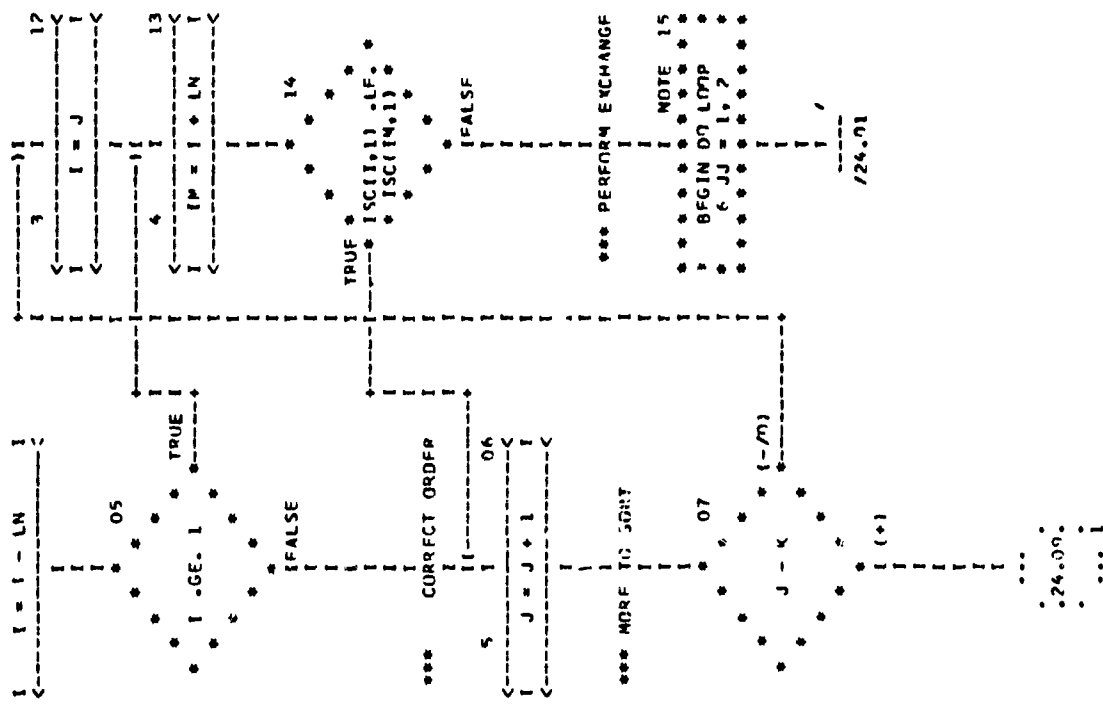
2562

297-24

Preceding page blank

-286-





07/22/71

AUTOFLOW CHART SET -

RPSS

PAGE 25

CHART TITLE - NON-PROCEDURAL STATEMENTS

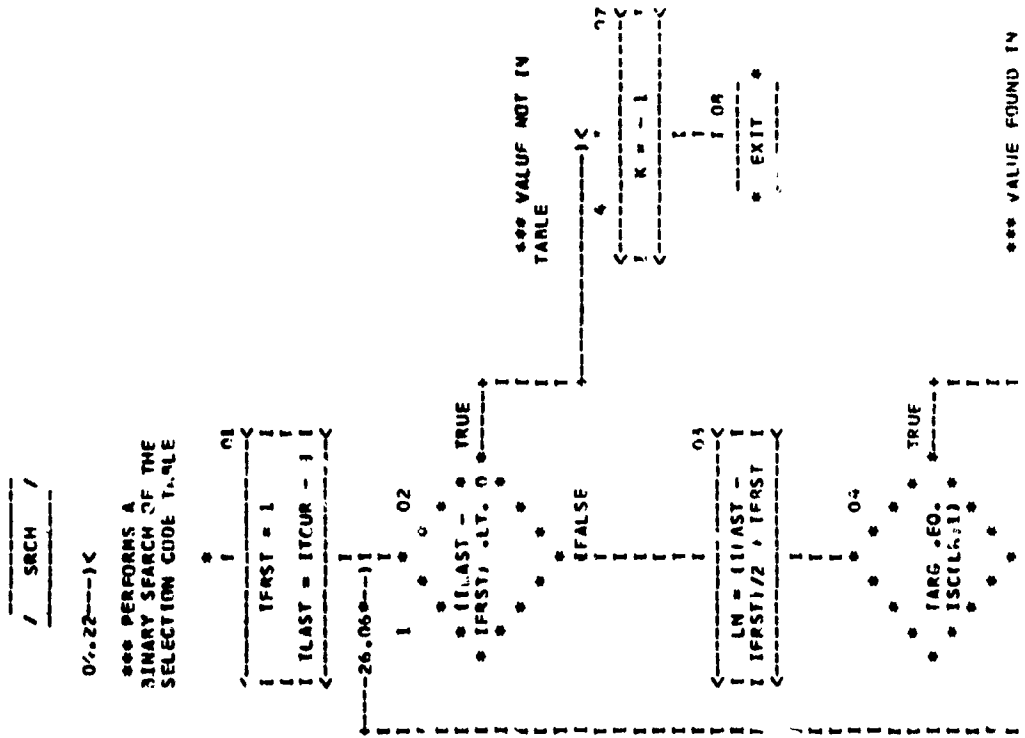
DIMENSION ISC(1000,2),I(1,1,2)
COMMON /A1/ 'SC,ITMAX,ITCUR,MINSC,MAXSC

07/22/71

AUTOFLOW CHART SET - RPSS

PAGE 26

CHART TITLE - SUBROUTINE SRCH(ARG,K)



Preceding page blank



07/22/71

CHART TITLE - NON-PROCEDURAL STATEMENTS

DIMENSION ISC(1000,2)
COMMON /A1/ ISC,ITHAX,ITCUR,MINSC,MAXSC

07/22/71

CHART TITLE - SURROUTINE INTWDL(K)

AUTOFLOW CHART SET -

PAGE 2A

INTWDL /

13.060--><

*** SETS UP THE
SELECTION CODE TABLE
ENTRIES
*** FIRST WORD
IS THE SELECTION CODE
*** SECOND
WORD BYTE 1 IS ZERO

BYTE 2 IS DENSITY AND
CHANNEL NUMBER

BYTE 3 IS DELETE FLAG

*** CLEAR TABLE
ENTRY

1	01
1	ISC(K,1) = 0
1	1
1	ISC(K,2) = 0
1	1

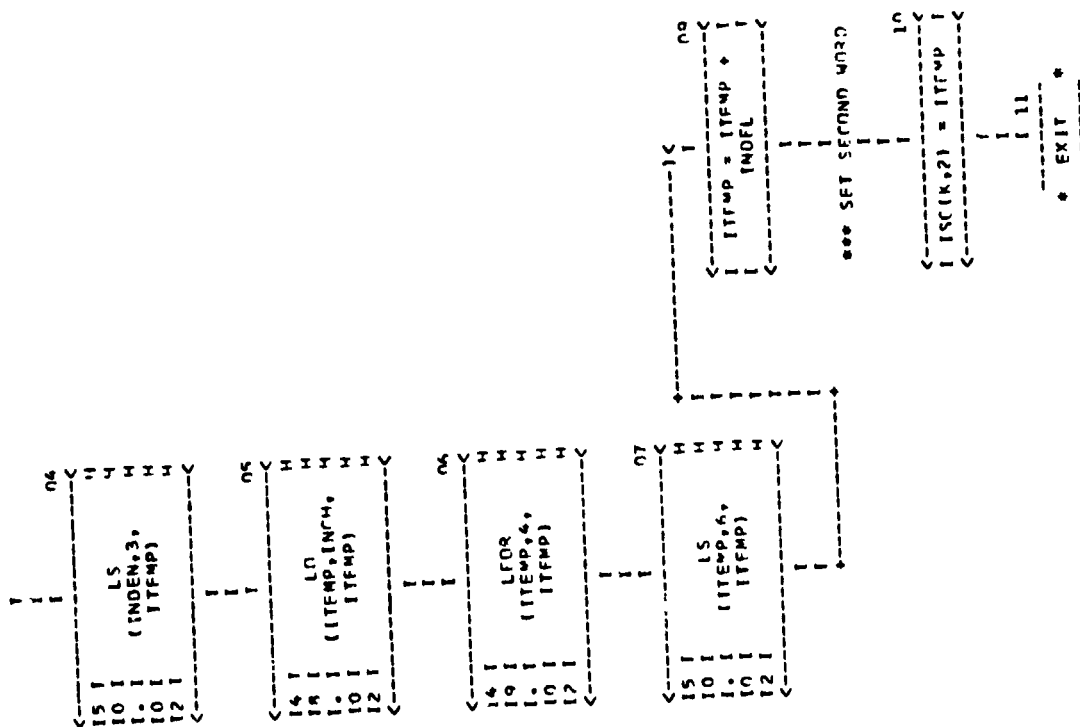
*** SET FIRST WORD

1	02
1	ISC(K,1) = IMSC
1	1

*** FORM SECOND BYTE
OF SECOND WORD

1	03
1	TEMP = 0
1	1

Preceding page blank



07/22/71

CHART TITLE - NON-PROCFOURAL STATEMENTS

AUTOFLOW CHART SET -

PPSS

PAGE 29

DIMENSION ISC(1000,7)

COMMON /A1/ ISC,ITMAX,ITCUR,MINSC,MAXSC

COMMON /B1/ INSC,INCH,INDEN,INDEL

/ OUTWOL /

14-06--1C

*** DECODE THE
SELECTION CODE TABLE
ENTRIES

*** OBTAIN SELECTION
CONF

$$\begin{array}{c} \bullet \\ \vdots \\ \hline Y \quad \text{INSC} = \text{ISC}(K,1) \quad \vdots \\ \hline \end{array}$$

*** STRIP SECOND
WFRD

```
<----->  
I I I  
<----->  
I TEMP = ISC(K,2) I  
<----->
```

LA	(TEMP, 1, INCH)	01
15	1	M
15	1	M
15	1	M
10	1	M
12	1	M

```

15 I
16 I
17 I
18 I
19 I
20 I
21 I
22 I
23 I
24 I
25 I
26 I
27 I
28 I
29 I
30 I
31 I
32 I
33 I
34 I
35 I
36 I
37 I
38 I
39 I
40 I
41 I
42 I
43 I
44 I
45 I
46 I
47 I
48 I
49 I
50 I
51 I
52 I
53 I
54 I
55 I
56 I
57 I
58 I
59 I
60 I
61 I
62 I
63 I
64 I
65 I
66 I
67 I
68 I
69 I
70 I
71 I
72 I
73 I
74 I
75 I
76 I
77 I
78 I
79 I
80 I
81 I
82 I
83 I
84 I
85 I
86 I
87 I
88 I
89 I
90 I
91 I
92 I
93 I
94 I
95 I
96 I
97 I
98 I
99 I
100 I
101 I
102 I
103 I
104 I
105 I
106 I
107 I
108 I
109 I
110 I
111 I
112 I
113 I
114 I
115 I
116 I
117 I
118 I
119 I
120 I
121 I
122 I
123 I
124 I
125 I
126 I
127 I
128 I
129 I
130 I
131 I
132 I
133 I
134 I
135 I
136 I
137 I
138 I
139 I
140 I
141 I
142 I
143 I
144 I
145 I
146 I
147 I
148 I
149 I
150 I
151 I
152 I
153 I
154 I
155 I
156 I
157 I
158 I
159 I
160 I
161 I
162 I
163 I
164 I
165 I
166 I
167 I
168 I
169 I
170 I
171 I
172 I
173 I
174 I
175 I
176 I
177 I
178 I
179 I
180 I
181 I
182 I
183 I
184 I
185 I
186 I
187 I
188 I
189 I
190 I
191 I
192 I
193 I
194 I
195 I
196 I
197 I
198 I
199 I
200 I
201 I
202 I
203 I
204 I
205 I
206 I
207 I
208 I
209 I
210 I
211 I
212 I
213 I
214 I
215 I
216 I
217 I
218 I
219 I
220 I
221 I
222 I
223 I
224 I
225 I
226 I
227 I
228 I
229 I
230 I
231 I
232 I
233 I
234 I
235 I
236 I
237 I
238 I
239 I
240 I
241 I
242 I
243 I
244 I
245 I
246 I
247 I
248 I
249 I
250 I
251 I
252 I
253 I
254 I
255 I
256 I
257 I
258 I
259 I
260 I
261 I
262 I
263 I
264 I
265 I
266 I
267 I
268 I
269 I
270 I
271 I
272 I
273 I
274 I
275 I
276 I
277 I
278 I
279 I
280 I
281 I
282 I
283 I
284 I
285 I
286 I
287 I
288 I
289 I
290 I
291 I
292 I
293 I
294 I
295 I
296 I
297 I
298 I
299 I
300 I
301 I
302 I
303 I
304 I
305 I
306 I
307 I
308 I
309 I
310 I
311 I
312 I
313 I
314 I
315 I
316 I
317 I
318 I
319 I
320 I
321 I
322 I
323 I
324 I
325 I
326 I
327 I
328 I
329 I
330 I
331 I
332 I
333 I
334 I
335 I
336 I
337 I
338 I
339 I
340 I
341 I
342 I
343 I
344 I
345 I
346 I
347 I
348 I
349 I
350 I
351 I
352 I
353 I
354 I
355 I
356 I
357 I
358 I
359 I
360 I
361 I
362 I
363 I
364 I
365 I
366 I
367 I
368 I
369 I
370 I
371 I
372 I
373 I
374 I
375 I
376 I
377 I
378 I
379 I
380 I
381 I
382 I
383 I
384 I
385 I
386 I
387 I
388 I
389 I
390 I
391 I
392 I
393 I
394 I
395 I
396 I
397 I
398 I
399 I
400 I
401 I
402 I
403 I
404 I
405 I
406 I
407 I
408 I
409 I
410 I
411 I
412 I
413 I
414 I
415 I
416 I
417 I
418 I
419 I
420 I
421 I
422 I
423 I
424 I
425 I
426 I
427 I
428 I
429 I
430 I
431 I
432 I
433 I
434 I
435 I
436 I
437 I
438 I
439 I
440 I
441 I
442 I
443 I
444 I
445 I
446 I
447 I
448 I
449 I
450 I
451 I
452 I
453 I
454 I
455 I
456 I
457 I
458 I
459 I
460 I
461 I
462 I
463 I
464 I
465 I
466 I
467 I
468 I
469 I
470 I
471 I
472 I
473 I
474 I
475 I
476 I
477 I
478 I
479 I
480 I
481 I
482 I
483 I
484 I
485 I
486 I
487 I
488 I
489 I
490 I
491 I
492 I
493 I
494 I
495 I
496 I
497 I
498 I
499 I
500 I
501 I
502 I
503 I
504 I
505 I
506 I
507 I
508 I
509 I
510 I
511 I
512 I
513 I
514 I
515 I
516 I
517 I
518 I
519 I
520 I
521 I
522 I
523 I
524 I
525 I
526 I
527 I
528 I
529 I
530 I
531 I
532 I
533 I
534 I
535 I
536 I
537 I
538 I
539 I
540 I
541 I
542 I
543 I
544 I
545 I
546 I
547 I
548 I
549 I
550 I
551 I
552 I
553 I
554 I
555 I
556 I
557 I
558 I
559 I
560 I
561 I
562 I
563 I
564 I
565 I
566 I
567 I
568 I
569 I
570 I
571 I
572 I
573 I
574 I
575 I
576 I
577 I
578 I
579 I
580 I
581 I
582 I
583 I
584 I
585 I
586 I
587 I
588 I
589 I
590 I
591 I
592 I
593 I
594 I
595 I
596 I
597 I
598 I
599 I
600 I
601 I
602 I
603 I
604 I
605 I
606 I
607 I
608 I
609 I
610 I
611 I
612 I
613 I
614 I
615 I
616 I
617 I
618 I
619 I
620 I
621 I
622 I
623 I
624 I
625 I
626 I
627 I
628 I
629 I
630 I
631 I
632 I
633 I
634 I
635 I
636 I
637 I
638 I
639 I
640 I
641 I
642 I
643 I
644 I
645 I
646 I
647 I
648 I
649 I
650 I
651 I
652 I
653 I
654 I
655 I
656 I
657 I
658 I
659 I
660 I
661 I
662 I
663 I
664 I
665 I
666 I
667 I
668 I
669 I
670 I
671 I
672 I
673 I
674 I
675 I
676 I
677 I
678 I
679 I
680 I
681 I
682 I
683 I
684 I
685 I
686 I
687 I
688 I
689 I
690 I
691 I
692 I
693 I
694 I
695 I
696 I
697 I
698 I
699 I
700 I
701 I
702 I
703 I
704 I
705 I
706 I
707 I
708 I
709 I
710 I
711 I
712 I
713 I
714 I
715 I
716 I
717 I
718 I
719 I
720 I
721 I
722 I
723 I
724 I
725 I
726 I
727 I
728 I
729 I
730 I
731 I
732 I
733 I
734 I
735 I
736 I
737 I
738 I
739 I
740 I
741 I
742 I
743 I
744 I
745 I
746 I
747 I
748 I
749 I
750 I
751 I
752 I
753 I
754 I
755 I
756 I
757 I
758 I
759 I
760 I
761 I
762 I
763 I
764 I
765 I
766 I
767 I
768 I
769 I
770 I
771 I
772 I
773 I
774 I
775 I
776 I
777 I
778 I
779 I
780 I
781 I
782 I
783 I
784 I
785 I
786 I
787 I
788 I
789 I
790 I
791 I
792 I
793 I
794 I
795 I
796 I
797 I
798 I
799 I
800 I
801 I
802 I
803 I
804 I
805 I
806 I
807 I
808 I
809 I
810 I
811 I
812 I
813 I
814 I
815 I
816 I
817 I
818 I
819 I
820 I
821 I
822 I
823 I
824 I
825 I
826 I
827 I
828 I
829 I
830 I
831 I
832 I
833 I
834 I
835 I
836 I
837 I
838 I
839 I
840 I
841 I
842 I
843 I
844 I
845 I
846 I
847 I
848 I
849 I
850
```

```

      I      I      05
      <----->
      15 I      LA      H
      14 I      (ITEMP, 54, H
      13 I      IODEN) H
      12 I      H
      11 I      H
      10 I      H
      9 I      H
      8 I      H
      7 I      H
      6 I      H
      5 I      H
      4 I      H
      3 I      H
      2 I      H
      1 I      H
      0 I      H
      <----->

      I      I      06
      <----->
      15 I      IODEN - I
      14 I      IODEN I
      13 I      IODEN I
      12 I      IODEN I
      11 I      IODEN I
      10 I      IODEN I
      9 I      IODEN I
      8 I      IODEN I
      7 I      IODEN I
      6 I      IODEN I
      5 I      IODEN I
      4 I      IODEN I
      3 I      IODEN I
      2 I      IODEN I
      1 I      IODEN I
      0 I      IODEN I
      <----->

      I      I      07
      <----->
      15 I      RS      H
      14 I      (IODEN, 3, H
      13 I      IODEN) H
      12 I      H
      11 I      H
      10 I      H
      9 I      H
      8 I      H
      7 I      H
      6 I      H
      5 I      H
      4 I      H
      3 I      H
      2 I      H
      1 I      H
      0 I      H
      <----->

      I      I      08
      <----->
      15 I      INCH - 4 I
      14 I      INCH - 4 I
      13 I      INCH - 4 I
      12 I      INCH - 4 I
      11 I      INCH - 4 I
      10 I      INCH - 4 I
      9 I      INCH - 4 I
      8 I      INCH - 4 I
      7 I      INCH - 4 I
      6 I      INCH - 4 I
      5 I      INCH - 4 I
      4 I      INCH - 4 I
      3 I      INCH - 4 I
      2 I      INCH - 4 I
      1 I      INCH - 4 I
      0 I      INCH - 4 I
      <----->

      I      I      09
      <----->
      15 I      EXIT
      14 I      EXIT
      13 I      EXIT
      12 I      EXIT
      11 I      EXIT
      10 I      EXIT
      9 I      EXIT
      8 I      EXIT
      7 I      EXIT
      6 I      EXIT
      5 I      EXIT
      4 I      EXIT
      3 I      EXIT
      2 I      EXIT
      1 I      EXIT
      0 I      EXIT
      <----->

```

07/22/71

CHART TITLE - NON-PROCEDURAL STATEMENTS

AUTOFLOW CHART SET -

RPSS

PAGE 31

DIMENSION ITC(11000,2)
COMMON /A1/ ISC,ITMAX,ITCUR,MINSC,MAXSC
COMMON /M1/ IOSC,IOCH,IOREN,IODEL

Preceding page blank

DATE /

```

02.098--J<

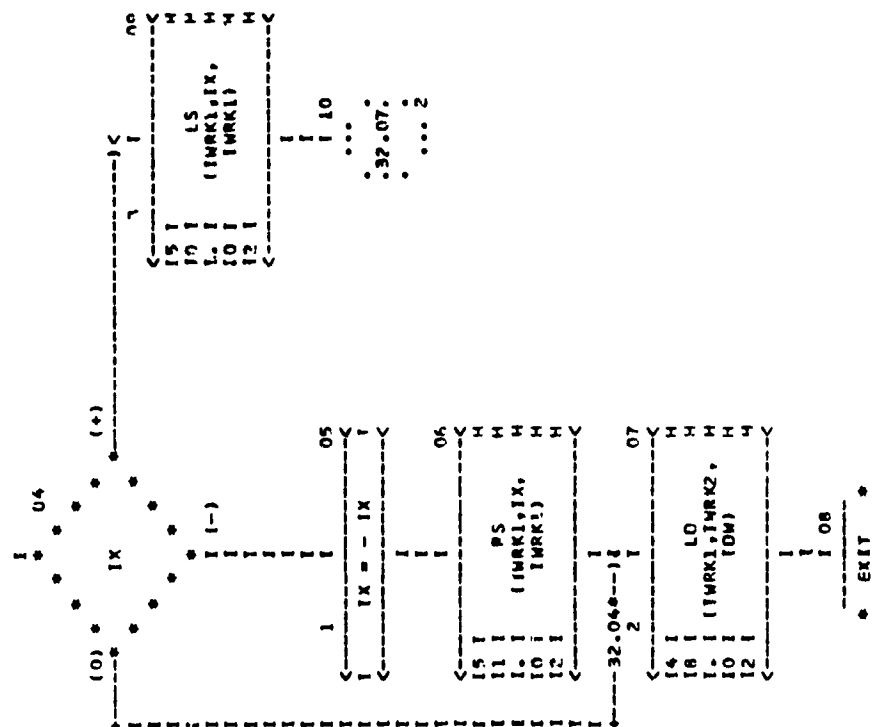
*** SHIFTS BYTES
FROM A GIVEN LOCATION
IN ONE WORD TO A
GIVEN
*** LOCATION IN
ANOTHER WORD
***
INPUT WORD IN =
***
BYTE LOCATION OF
INPUT WORD TO BE
SHIFTED - 1,2 OR 3
***
OUTPUT WORD
***
FOR =
BYTE LOCATION TO BE
SHIFTED INTO IN
OUTPUT WORD-1,2,3

```

	01	H
(I	H
17 I		H
18 I	IA	H
19 I	(IM,MASK(II),	H
10 I	IMP(KI)	H
12 I)	H

15 1
15 1
15 1
10 1
(2 1
LA
(10W
MASKZ(10R1
IMRK2)

IX - 45 (194)



07/22/71

CHART TITLE - NON-PROCEDURAL STATEMENTS

```

DIMENSION MASK1(3),MASK2(3),MASK5(6)
COMMON /MASK/ MASK1,MASK2
EQUIVALENCE (MASK1(1),MASK1(1))
    
```

07/22/71

AUTOFLOW CHART SET -

PAGE

PAGE 34

CHART TITLE - SUBROUTINE IMESG

Preceding page blank

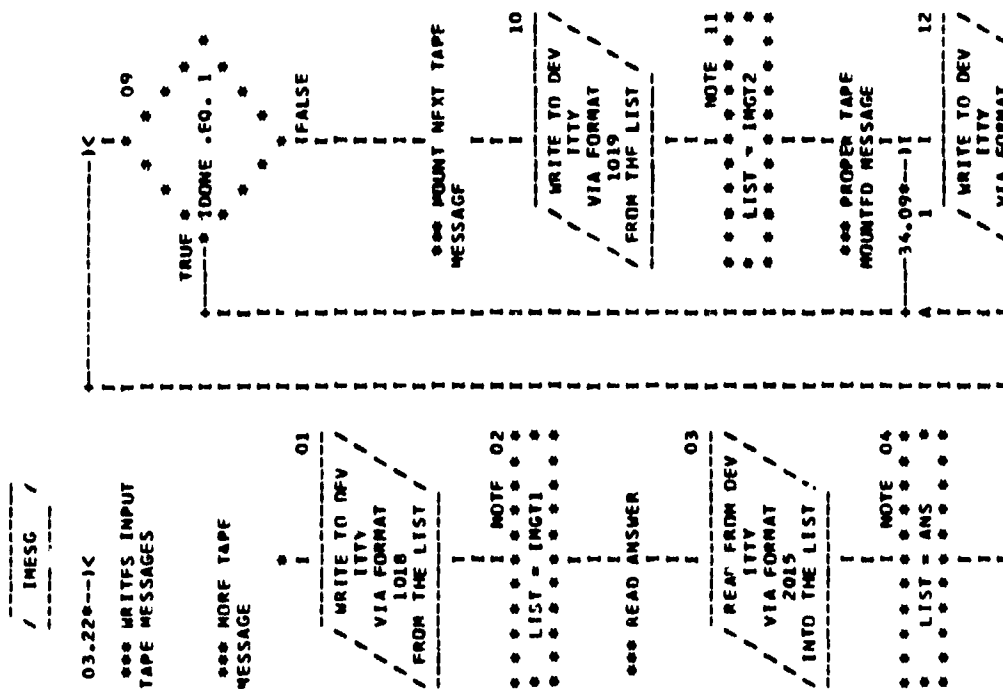




CHART TITLE - NON-PROCEDURAL STATEMENTS

```
COMMON /E1/ TM,TA,TF,TX,TY, \J,TP,TT,TS,TP,TH
COMMON /F1/ ITTY,IPT,IDI,INP,IPRTP
COMMON /G1/ IDR1,IDR2,INW3,INDEC
COMMON /J1/ INGT1,INGT2,INONE
1018  FORMAT (54H ARE THERE MORE INPUT TAPES FOR THIS RUN IN ADDITION T,
      64H THAT/ 27H CURRENTLY MOUNTED ON DRIVE,IX,IL,15H  ENTER Y OR N,
2015  FORMAT (A1)
1019  FORMAT (32H MOUNT NEXT INPUT TAPE ON DRIVE ,I1)
1020  FORMAT (50H ENTER D IF PROPER INPUT TAPE IS MOUNTED ON DRIVE ,I1)
```

1
1
2

/ NPAGE /

*** OUTPUTS THE
HEADING INFORMATION
FOR THE SELECTION
CODE LIST

1 01

/ WRITE IN DEV /
/ 1PRTS /
/ VIA FORMAT /
/ 1013 /
/ FROM THE LIS /

```

      * * * * * I   I   NOTF    02
      * * * * * LIST = MONTH+
      * * * * * IDAY, IYEAR,
      * * * * * IPAGE
      * * * * *

```

1
1
1

03

WRITE TO DEV
IPATR
VIA FORMAT
1014
FROM THE LIST

Y Y Y NOV 1961

1
1
1

05

WRITE TO DEV
IPTR
VIA FORMAT
1015

CHART TITLE - NON-PROCEDURAL STATEMENTS

	COMMON /F1/	ITTY, IPT, IDT, INP, IPRTR	
	COMMON /F1/	IMONTH, IDAY, IYFAR, IPAGE, IFILE, IREV	
1013	FORMAT (47H1	RPS SELECTION CODE TABLE - INPUT SUMMARY	*I2, I4/,
	12, I4/, I2, I0H	PAGE	, I2)
1014	FORMAT (14H0	TAPE NUMBER	, I5)
1015	FORMAT (14H	FILE NUMBER	, I2)
1016	FORMAT (14H	REVISION	, I2)
1017	FORMAT (53H0	SELECTION CODE	CHANNEL
	6HE FLAG)	DEVSITY	DELET,

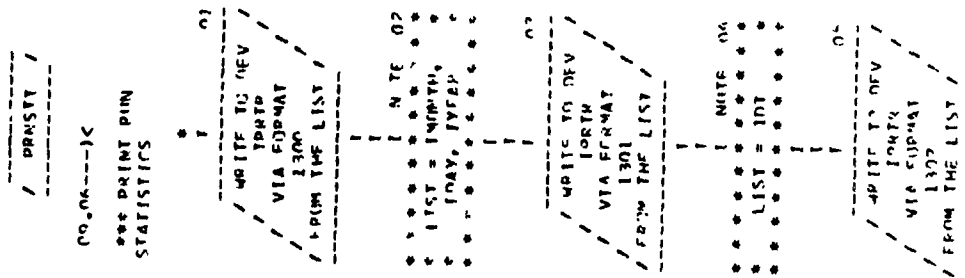
07/22/71

CHART TITLE - SUBROUTINE PRNSTT

AUTOFLOW CHART SET -

PPSS

Page 29



Reproduced from
best available copy.

Preceding page blank

-315-

07/22/71

CHART TITLE - NON-PROCEDURAL STATEMENTS

AUTOFLW CHART SET - RPSS

PAGE 19

```

1300 DIMENSION ICNT(25),ICH(4),ITRUF(740)
1301 COMMON /K1/ ICNT,ICH
1302 COMMON /TRUF/ ITRUF,NYS,NXS
1303 COMMON /L1/ IWIN,ISFC,IOFF
1304 COMMON /F1/ ITTY,IPY,INT,INP,IPRTP
1305 COMMON /I1/ IMONTH,ICAY,IYEAR,IPAGE,IFILT,IPREV
1306 FORMAT (24H1 ROS RUN STATISTICS OF ,I2,I4,I2,I4,I2)
1307 FORMAT (14H1 TAPE NUMBER ,I5)
1308 FORMAT (14H1 FILE NUMBER ,I2)
1309 FORMAT(14H1 RUN STATISTICS//14H ELAPSED TIME = ,I4,I4,I2HMINUTES A
1310 ND ,I4,I4,I2HSECONDS//14,I4,I2HSCAN LINES INPUT CONTAINING ,I4,I4,
1311 14H X COORDINATES//13PH X DESIGNATOR OF POINTS PROCESSED WERE//12
1312 5,I4HCODE,I2H,SUMOUNT,25I/25X,I4,I2X,I6)
1313 FORMAT (30H0CHANNEL TOTALS CHANNEL 0 = ,I4,I4,I2H CHANNEL 1 = ,I
1314 4,I4,I2H CHANNEL 2 = ,I4,I4,I2H CHANNEL 3 = ,I4)

```


CHART TITLE - SURPLUTINE NSCT

/ NSFT /

03.234-12

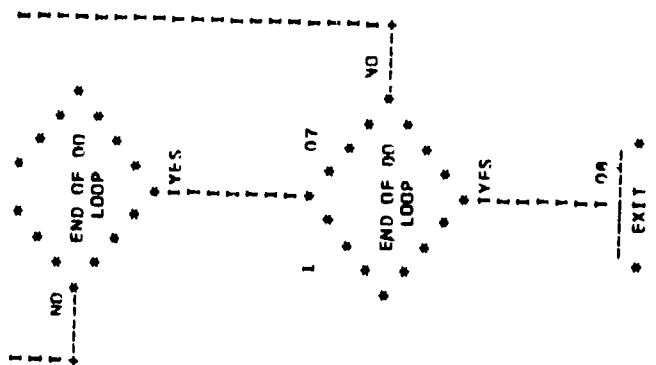
```

*** DESITS OUTPUT
*** BUFFER PRINTERS AND
*** CLEARS THE BUFFER TO
*** THE HIGH
*** IGNORE VALUE

```

	1	01
<	I	<
I	NENT = 1	I
I		I
I	LSTOVT = 1	I
I		I
I	LSTORU = ?	I
I		I
>		>

[illegible][illegible]



07/22/71

CHART TITLE - NON-PROCEDURAL STATEMENTS

AUTOFLOW CHART SET -

RPSS

PAGE 41

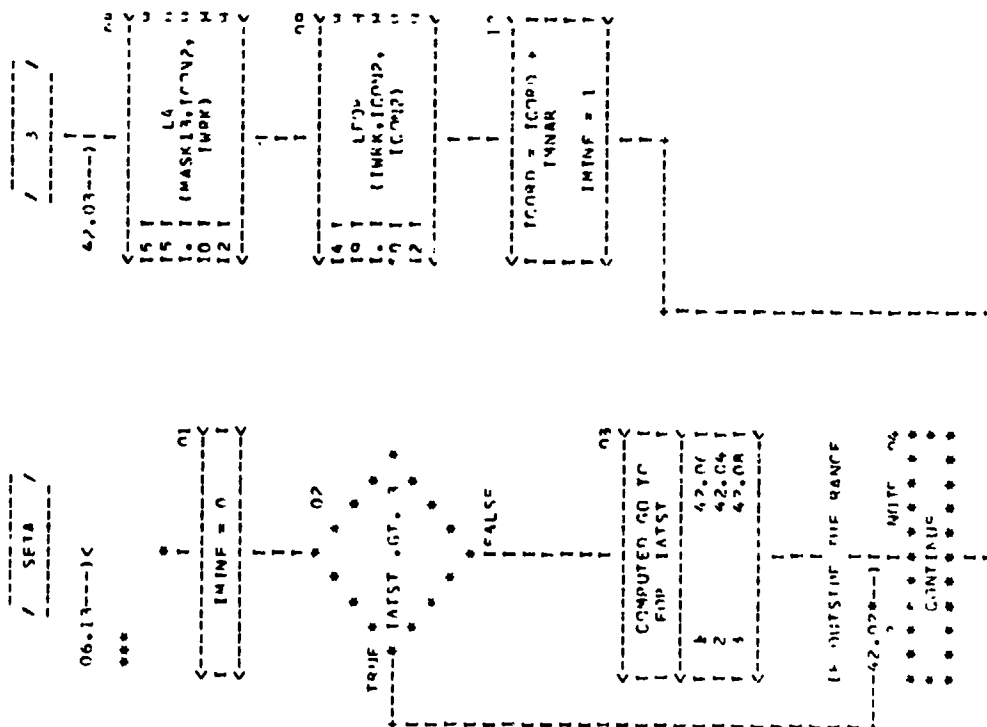
DIMENSION INSET(4), IORUF(1364)
COMMON /M1/ INSET, LSTWRD, LSTBYT, NENT, IORUF

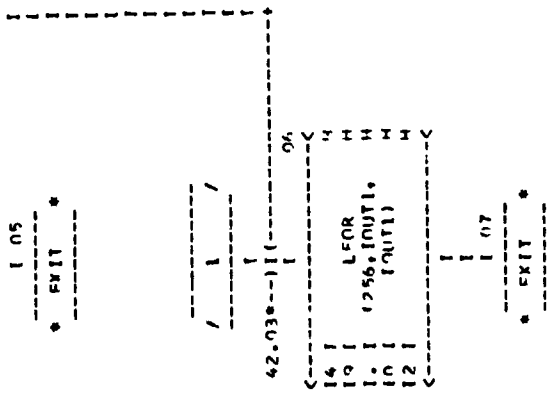
07/22/71

CHART TITLE - SUBROUTINE SETA

AUTOFLOW CHART SFT -

PAGE 42





07/22/71

AUTOFLOW CHART SET -

RPSS

PAGE 43

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
DIMENSION MASK1(3),MASK2(3)
COMMON /GNRI/ JTYPE,ICORD,ICON1,ICON2,IX,IX,IFT,ISTAT
COMMON /AFIC/ INTNF,IMNAR
COMMON /MASK/ MASK1,MASK2
COMMON /IOUT/ IOUT1,IOUT2,IATST
EQUIVALENCE (MASK13,MASK1(3))
```

Preceding page blank

07/22/71

CHART TITLE - PLOCH DATA

AUTOFLOW CHART SET -

PPSS

PAGE 44

Preceding page blank

-326-

07/22/71

CHART TITLE - NON-PROCEDURAL STATEMENTS

```

COMMON /EL/ IN,TA,TF,IX,IV,TD,TP,TT,TS,TR,YM
DATA IV,TA,TF/SHN ,SHA ,SHF \ /
DATA IX,IV,TD,TP/SHC ,SHY ,SHN ,SHO /
DATA TT,TS,TP,TY/SHT ,SHS ,SHW ,SHM /
    
```


07/22/71

CHART TITLE - COMPUTING SETM(1)

AUTOFLIM CHART SET -

3PSS

PAGE 44

1 SETM 1

02.07---)C

THIS IS AN ASSEMBLY
LANGUAGE ROUTINE

1 01
EXIT

07/22/71

CHART TITLE - SUBROUTINE SETC(1)

/ SETC /

C2.0A---)<

THIS IS AN ASSEMBLY
LANGUAGE ROUTINE

*
1 02

* EXIT *

07/22/71

~ HAPT TITLE - SUBROUTINE LOIT,J,<)

AUTOFLOW CHART SET -

PAGE 4A

/ LO /

03.26*--)<

THIS IS AN ASSEMBLY
LANGUAGE ROUTINE

*
T 02

* EXIT *

07/22/71

CHART TITLE - SUBROUTINE FOR (I,J,K) 1

AUTOFLOW CHART SET -

PDSS

147 40

/ FOR /

03.278-->C

THIS IS AN ASSEMBLY
LANGUAGE ROUTINE

*
I 02

* EXIT *

07/22/71

CHART TITLE - SUPROUTINE LS(I,J,K)

/ LS /

C2.12*--!<

THIS IS AN ASSEMBLY
LANGUAGE ROUTINE

*
I 02

* EXIT *

07/22/71

CHART TITLE - SUBROUTINE RSIT,J,K

/ RS /

02.110---)C

THIS IS AN ASSEMBLY
LANGUAGE ROUTINE

*
I 02

* EXIT *

07/22/71

CHART TITLE - SUBROUTINE MCGRWIT,J,K,L,M)

/ MCGRW /

07.06---)<

THIS IS AN ASSEMBLY
LANGUAGE ROUTINE

*
I 02

* EXIT *

07/22/71

CHART TITLE - SUBROUTINE REOF(11)

/ REOF /

19.09---)<
THIS IS AN ASSEMBLY
LANGUAGE ROUTINE

*
1 02

* EXIT *

07/22/71

AUTOFLOW CHART SET -

3 PSS

PAGE 94

CHART TITLE - SUBROUTINE TIME(I,J,K)

/ TIME /

04.170--)<

THIS IS A SYSTEM
SUBROUTINE

*
1.02

* EXIT *

07/22/71

CHART TITLE - SUBROUTINE LALL,J,K)

AUTOFLOW CHART SET -

HPSS

PAGE 99

/ LA /

04.20*-1<

THIS IS AN ASSEMBLY
LANGUAGE SUBROUTINE

*
T 02

* EXIT *